Trigger di Schmitt Robot Golem Registri come è fatto un PC
Corso Verilog comunicazione allarme SMS ricevitore morse
RS485 radiocomando filtri attivi gestione pompe con Arduino

www.farelettronica.com
312 GIUGNO Mensile Anno XXVII € 6,00

UN ALLARME SMS PER LA TUA AUTO

Gestione di elettropompe con ARDUINO

RADIOCOMANDO A 8 CANALI

COMUNICAZIONE DATI IN RS485

RICETRASMETTITORE MORSE

SCOPRI COME È FATTO UN COMPUTER

SIMULATORE DI PRESENZA

ISSN 1591-2272
10312
9 771591 227008

Per scoprire le novità!
digikey.it/new
Digi-Key CORPORATION

Firmware

l'unica rivista italiana nativa digitale per i professionisti dell'elettronica e dedicata, principalmente, ai micontrontrollori, dispositivi FPGA, componentistica analogica e approfondimenti sulle tecnologie

- non più solo testo ma anche video!
- possibilità di grande interazione per il lettore
- links a tutte le varie risorse aggiuntive
- possibilità di cercare un testo nella rivista corrente e nell'archivio (!)
- possibilità di stampare tutta la rivista o anche solo alcune parti
- possibilità di leggere la rivista offline scaricandola sul PC
- possibilità di leggere la rivista con gli e-reader (compreso iPhone e iPad)
- moltissime riviste in archivio GRATIS per i nuovi abbonati
- abbonamento a partire da € 19.50

Richiedi la tua copia omaggio qui:
http://mailing.fwonline.eu

CODICE MIP 2808405

LA RICERCA DEL VOSTRO PRODOTTO È DIVENTATA PIÙ FACILE.

NUOVO MOTORE DI RICERCA: ORDINA PER PREZZO I RISULTATI!

SPEDIZIONE GRATUITA
PER ORDINI DI OLTRE € 65!*

Digi-Key® CORPORATION

Con questa nuova funzione, i clienti possono ordinare per prezzo in ordine crescente o decrescente oltre ad applicare dei filtri per il prodotto in magazzino, il prodotto senza piombo, ed il prodotto compatibile RoHS.

La scelta più ampia di prodotti nel settore, disponibili per la spedizione immediata.

www.digikey.it
800.786.310

*A tutti gli ordini di importo inferiore a € 65,00 sarà aggiunto un addebito per la spedizione di importo pari a € 18,00. Tutti gli ordini vengono spediti tramite UPS, consegna entro 1-3 giorni (secondo la destinazione finale). Nessun addebito per i costi di imballaggio. Tutti i prezzi sono in euro e comprensivi di imposte. Se peso eccessivo o circostanze eccezionali dovessero comportare un addebito diverso, i clienti verranno contattati prima della spedizione dell'ordine. Digi-Key è un distributore autorizzato di tutti questi fornitori. Ogni giorno vengono aggiunti nuovi prodotti. © 2011 Digi-Key Corporation, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

CODICE MIP 2800828

CONNETTITI

sommario

312 *giugno 2011*

Zoom in

20 COME È FATTO UN PC?

La scheda madre è il circuito elettronico più importante in un PC. Viene appunto chiamata "madre" (motherboard o mainboard) perché tutto, all'interno del PC, vi è collegato e tutto deve comunicare con essa affinché il PC possa funzionare. Analizziamola in dettaglio insieme ai suoi componenti principali.

di Maurizio Di Paolo Emilio

Progettare & costruire

26 GESTIONE DI ELETTROPOMPE CON ARDUINO

Una curiosa ed utile applicazione di Arduino per assicurare i vostri locali da possibili allagamenti…

di Andrea ed Enrico Tobaldo

32 ALLARME SMS PER AUTO

Un sistema di antifurto per automobili in grado di avvertire il proprietario del mezzo del tentativo di furto, ma non solo! Il circuito dovrà essere in grado di segnalare la posizione dell'auto ovunque questa si trovi. Cosa c'è di meglio allora che sfruttare la rete cellulare GSM che copre quasi la totalità del territorio nazionale?

di Luca Stanzani

38 DISSUASORE AUDIO CON MESSAGGI VOCALI

Dissuasore audio abbinabile a qualsiasi sensore a doppia tecnologia. Attivabile e disattivabile tramite centrale di allarme, con uscita di attivazione allarme e di relè per lampeggiante, possibilità d'uso anche (da solo) senza centrale d'allarme.

di Giuseppe La Rosa

Imparare & approfondire

48 FILTRI ATTIVI CON AMPLIFICATORI OPERAZIONALI TERZA PARTE

La trattazione continua in questa terza ed ultima puntata affrontando due argomenti molto importanti: i filtri selettivi ed i filtri elimina banda (notch).

di Grazia Ancona

58 CORSO DI PHYTHON PARTE I VALUTAZIONE DELLE ESPRESSIONI

Come in qualsiasi linguaggio di programmazione, anche in Python è possibile espressioni che l'interprete valuta in tempo reale e ne visualizza il risultato. Vediamo in dettaglio come funziona questo meccanismo.

di Massimiliano Miocchi

Rispondi e... **VINCI!** *pag.* **68**

DIVERTITI E METTI ALLA PROVA LE TUE CONOSCENZE CON **ELETTRO QUIZ** E VINCI OGNI MESE ESCLUSIVI PREMI!

GLI ARTICOLI CONTRASSEGNATI COL SIMBOLO WWW sono già disponibili in formato PDF* all'indirizzo www.farelettronica.com/club

*Puoi iscriverti al CLUB di Fare Elettronica versando una piccola quota annuale.

62 LA COMUNICAZIONE IN RS485
Una trattazione per studiare come avviene la comunicazione RS485 tra dispositivi a microcontrollore PIC, utilizzando il mikroBasic PRO.
di Giovanni Di Maria

70 IL TRIGGER DI SCHMITT
Il funzionamento del Trigger di Schmitt: un circuito per la realizzazione di soglie di tensione e impiegato in tutti i circuiti che richiedano di lavorare con valori variabili in ingresso.
di Davide Boscolo

74 CORSO SYSTEMVERILOG
PARTE SECONDA
PROCEDURE, TASK E FUNZIONI
I blocchi procedurali sono insiemi di istruzioni eseguite in seguito all'attivazione del blocco stesso. Eccom come crearli e richiamarli.
di Mariano Severi

L'angolo di Mr A.Keer

78 I REGISTER FILE
Continua in questa puntata la descrizione dei componenti correlati con l'elemento di memoria di tipo D; la struttura del Registro, fondamentale nel progetto digitale, viene ora organizzata in batteria, al fine di disporre di dispositivi che possano essere scritti e letti contemporaneamente.
di Giorgio Ober

Radio & radio

88 MINI RICETRASMETTITORE MORSE
Un semplice RTX da costruire nei ritagli di tempo che vi permetterà di esercitarvi nell'antica arte delle trasmissioni in codice Morse.
di Remo Riglioni

92 RTX 8 CANALI: IL TRASMETTITORE
Di radiocomandi ne sono stati presentati molti nelle pagine delle varie riviste del settore, ma ben pochi presentano le caratteristiche di semplicità e praticità del progetto che stiamo presentando.
di Silvano Breggion

Robot Zone

102 ROBOT GOLEM
PARTE QUARTA
LA "TESTA"
In questo articolo vedremo in dettaglio la costruzione della testa del robot, installandovi le webcam, i microfoni e l'altoparlante.
di Mattias Costantini

rubriche

7 Editoriale
10 Idee di progetto
14 News
19 Eventi
68 Elettroquiz
108 IESHOP

ABBIAMO BISOGNO DEL TUO AIUTO!

FACCI CONOSCERE LE TUE PREFERENZE VOTANDO IL TUO ARTICOLO PREFERITO SUL SITO WWW.FARELETTRONICA.COM

elenco inserzionisti

Digi-Key Corporation Pag. 1,3
701 Brooks Ave South - 56701 Thief River Falls, MN
Tel. 800*338*4105 x1454 - www.digikey.it

Elettroshop Pag. 65
Via Giotto, 7 - 20032 Cormano (MI)
Tel. 02 66504755 - www.elettroshop.com

Evr Pag. 94
Viale Kennedy, 96 - 20027 Rescaldina (MI)
Tel. 0331 1815404 - www.evr-electronics.com

Farnell Italia Pag. 6-31
Corso Europa 20 - 22 - 20020 Lainate (MI)
Tel. 02 93995(1) - www.farnell.com

Futura Elettronica Pag. 25
Via Adige 11 - 21013 Gallarate (VA)
Tel. 0331-799775 - www.futuranet.it

LeCroy Pag. 13
via E. Mattei Valecenter 1/c/102a - 30020 Marcon (VE)
Tel. 041/5997011 - www.lecroy.com

Microchip Technology Pag. 9
Via Pablo Picasso, 41 - 20025 Legnano (MI)
Tel: +39 0331 74261 - www.microchip.com

Micromed Pag. 60
Via Valpadana 126B/2 - 00141 Roma (RM)
Tel. 06 9058496 - www.micromed.it

MikroElektronika Pag. 43
Visegradska, 1A - 11000 Belgrade
Tel. +381 11 3628830 - www.mikroe.com

Millennium Dataware Pag. 35
Corso Repubblica 48 - 15057 Tortona (AL)
Tel. 0131-860254 - www.mdsrl.it

PCB-Pool Pag. 19
Bay 98-99 - Shannon Free Zone
Shannon - County Clare
Tel. 02 64672645 - www.pcb-pool.com

R.C.C. Pag. 15
Via G. Di Vittorio 19 - 20097 San Donato Milanese (MI)
Tel. 02-51876194 - www.rccitaly.com

Tecnoimprese Pag. 107
Via Console Flaminio, 19 - 20134 (MI)
Tel. 02 210.111.1 - www.fortronic.it

Teltools Pag. 97
Via Della Martinella 9 - 20152 Milano (MI)
www.teltools.it

LUPUS IN FABULA

Anritsu, 14 - Arduino, 26 - Atmel, 32 - Aurel, 92 - Erni Electronics, 15 - Mentor Graphics, 15 - Microchip, 38, 62, 92 MSC, 14 - OMP Mechtron, 14 - Python, 58 Telecontrolli, 92

LE TECNOLOGIE PIÙ RECENTI...
SUBITO DISPONIBILI DA FARNELL

> Oltre 100 nuove tecnologie aggiunte online ogni giorno

> La gamma più ampia della distribuzione europea con 420.000 prodotti da 3.500 brand leader

> element14: la prima comunità virtuale per progettisti elettronici

> eQuotes: ricevi le tue quotazioni e confermale online con un click

> i-Buy: la soluzione per l'eProcurement gratuita da Farnell

www.farnell.com/it

CODICE MIP 2807444

element14

www.element14.com

Farnell

Progetta con i migliori

A Premier Farnell Company

di MAURIZIO DEL CORSO

SMS antifurto

Il problema dei furti nelle auto è molto comune in tutto il mondo. Pensate a come sarebbe utile se la nostra auto ci avvisasse con un SMS che sta per essere scassinata!

Negli anni '80 l'incubo degli automobilisti era il furto dell'autoradio. I vecchi modelli a cassette potevano essere estratti, ma spesso per non portarsi dietro un pezzo così ingombrante lo si preferiva nasconderlo sotto il sedile. Oggi i tempi sono molto cambiati: le autoradio sono per lo più già integrate nell'autovettura e, quando non lo sono, è possibile estrarre solo la mascherina. Ma anche le esigente dei potenziali ladri sono cambiate: oggi sono molto appetibili i navigatori portatili e, a seconda dei modelli, anche l'intera autovettura! Per difendere la nostra auto vi proponiamo un progetto piuttosto complesso ma molto interessante per imparare qualcosa di nuovo: un antifurto con modulo GSM in grado di inviarvi un SMS se la portiera della vostra auto viene aperta da qualche malintenzionato. E' vero, di antifurti se ne trovano a migliaia in commercio anche a prezzi veramente accessibili, ma capire effettivamente come poter inviare un SMS in modo automatico sfruttando un modulo GSM embedded, davvero non ha prezzo!

in fatto di "campagne"... la nostra non ha rivali

ABBONATI SUBITO

FINO AL 40% DI SCONTO !!!

Scopri gli altri abbonamenti in offerta qui
abbonamenti.ieshop.it
e la PROMOTUTTE3

fare elettronica

DIRETTORE RESPONSABILE
Antonio Cirella

DIRETTORE TECNICO
Maurizio Del Corso

Segreteria di redazione
Giorgia Generali

Comitato Scientifico
Simone Masoni (Microtest), Francesco Picchi (Microtest), Massimo Rovini (Università degli Studi di Pisa).

Art Director
Patrizia Villa

Hanno collaborato in questo numero:
Grazia Ancona, Davide Boscolo, Silvano Breggion, Mattias Costantini, Francesco Di Lorenzo, Giovanni Di Maria, Maurizio Di Paolo Emilio, Antonio Giannico, Giuseppe La Rosa, Massimiliano Miocchi, Giorgio Ober, Remo Riglioni, Mariano Severi, Luca Stanzani, Enrico Tobaldo.

Direzione Redazione
INWARE Edizioni srl
Via Giotto, 7
20032 Cormano (MI)
Tel. 02.66504755
Fax 02.66508225
info@inwaredizioni.it
www.inwaredizioni.it
Redazione: fe@inwaredizioni.it

Pubblicitá per l'Italia
Agostino Simone
Tel. 347 2230684
media@inwaredizioni.it

International Advertisement
Elisabetta Rossi
Tel. 328 3245956
international@inwaredizioni.it

Stampa
Grafiche Sima
Via Provinciale Francesca, 52
24040 Ciserano (BG)

Distribuzione
Parrini & C s.p.a.
Via di Santa Cornelia, 9
00060 Formello (RM)

Ufficio Abbonamenti
INWARE Edizioni srl
Via Giotto, 7
20032 Cormano (MI)
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@inwaredizioni.it
Tel. 02.66504755
Fax. 02.66508225
L'ufficio abbonamenti è disponibile telefonicamente dal lunedì al venerdì dalle 14,30 alle 17,30.
Tel. 02.66504755
Fax 02.66508225

Abbonamento per l'Italia:
€ 55,00
Abbonamento per l'estero:
€ 115,00
Gli arretrati potranno essere richiesti, per iscritto, a **€ 9,00** oltre le spese di spedizione

Autorizzazione alla pubblicazione
Tribunale di Milano n.647 del 17/11/2003

USPI
Mensile associato all'USPI (Unione Stampa Periodica Italiana)

© Copyright
Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti, disegni e fotografie sono di proprietà di Inware Edizioni srl. È vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

Privacy
Nel caso la rivista sia pervenuta in abbonamento o in omaggio, si rende noto che i dati in nostro possesso sono impiegati nel pieno rispetto del D.Lgs. 196/2003. I dati trasmessi a mezzo cartoline o questionari presenti nella rivista, potranno venire utilizzati per indagini di mercato, proposte commerciali, o l'inoltro di altri prodotti editoriali a scopo di saggio. L'interessato potrà avvalersi dei diritti previsti dalla succitata legge. In conformità a quanto disposto dal Codice di deontologia relativo al Trattamento di dati personali art. 2, comma 2, si comunica che presso la nostra sede di Cormano Via Cadorna 27, esiste una banca dati di uso redazionale. Gli interessati potranno esercitare i diritti previsti dal D.Lgs. 196/2003 contattando il Responsabile del Trattamento Inware Edizioni Srl (info@inwaredizioni.it).

Collaborare con FARE ELETTRONICA
Le richieste di collaborazione vanno indirizzate all'attenzione di Maurizio Del Corso (m.delcorso@inwaredizioni.it) e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

Microcontroller a basso consumo per progetti "battery-friendly"

Microchip offre le correnti più basse in modalità attiva e sleep

Maggiore durata delle batterie grazie ai microcontroller PIC® con tecnologia nanoWatt XLP: le correnti più basse in modalità Active e Sleep oggi disponibili sul mercato.

Le nuove famiglie Microchip PIC12F182X, PIC16F182X e PIC16F19XX sono particolarmente ricche di periferiche e offrono correnti operative inferiori a 50 µA e correnti di sleep fino al limite inferiore di 20 nA. Questi prodotti permettono di dare vita a progetti battery-friendly che possono incorporare funzionalità di rilevamento tattile capacitivo, di visualizzazione LCD, di comunicazione e molto altro ancora per differenziare il vostro prodotto.

L'architettura potenziata a 8-bit di fascia media ideata da Microchip offre un incremento del 50% in termini di prestazioni e 14 nuove istruzioni che garantiscono un miglioramento superire al 40% nell'esecuzione del codice rispetto agli MCU PIC16 a 8-bit della generazione precedente.

Le famiglie PIC12F182X e PIC16F182X prevedono:

- Package da 8 a 64 pin
- Touch-sensing capacitivo mTouch™
- Varie periferiche di comunicazione
- Interfaccia doppia I²C™/SPI
- Uscite PWM con base tempi indipendenti
- Modulatore segnale dati

La famiglia PIC16F19XX prevede:

- Touch-sensing capacitivo mTouch™
- Pilotaggio LCD
- Varie periferiche di comunicazione
- Più canali PWM con timer indipendenti
- Fino a 28 KB di memoria programma Flash
- EEPROM dati potenziata
- Riferimento a 32livelli
- Tre comparatori d'ingresso rail-to-rail

3 SEMPLICI MOSSE PER INIZIARE

1. Esaminate il video Low Power Comparison
2. Scaricate le note e i suggerimenti relativi alle applicazioni low-power
3. Ordinate le campionature e i tool di sviluppo al link www.microchip.com/XLP

PIC16F193X 'F1' Evaluation Platform - DM164130-1

Microcontrollers | Digital Signal Controllers | Analog | Memory | RF & Wireless

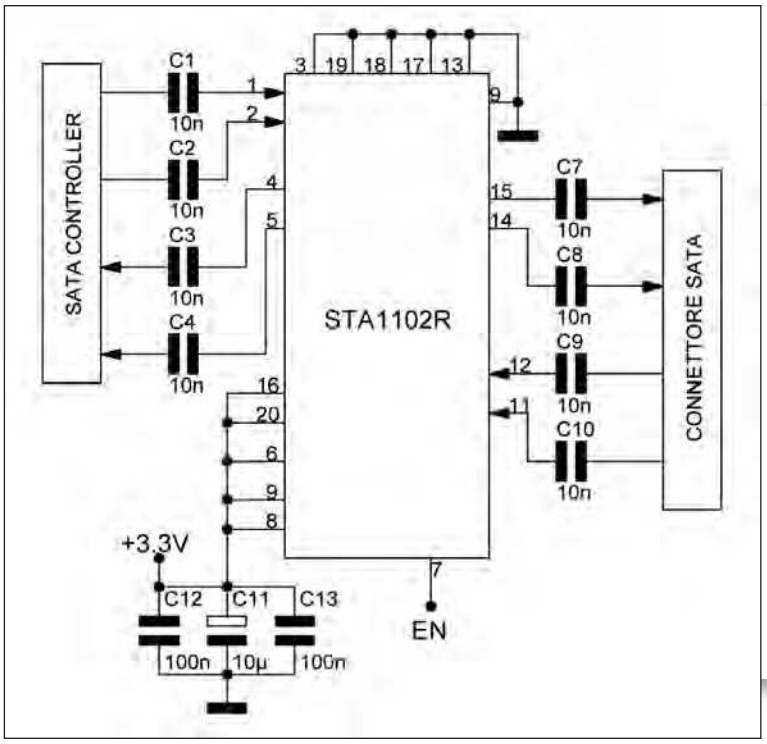
CODICE MIP 2806836

microchip DIRECT
www.microchipdirect.com

www.microchip.com/xlp

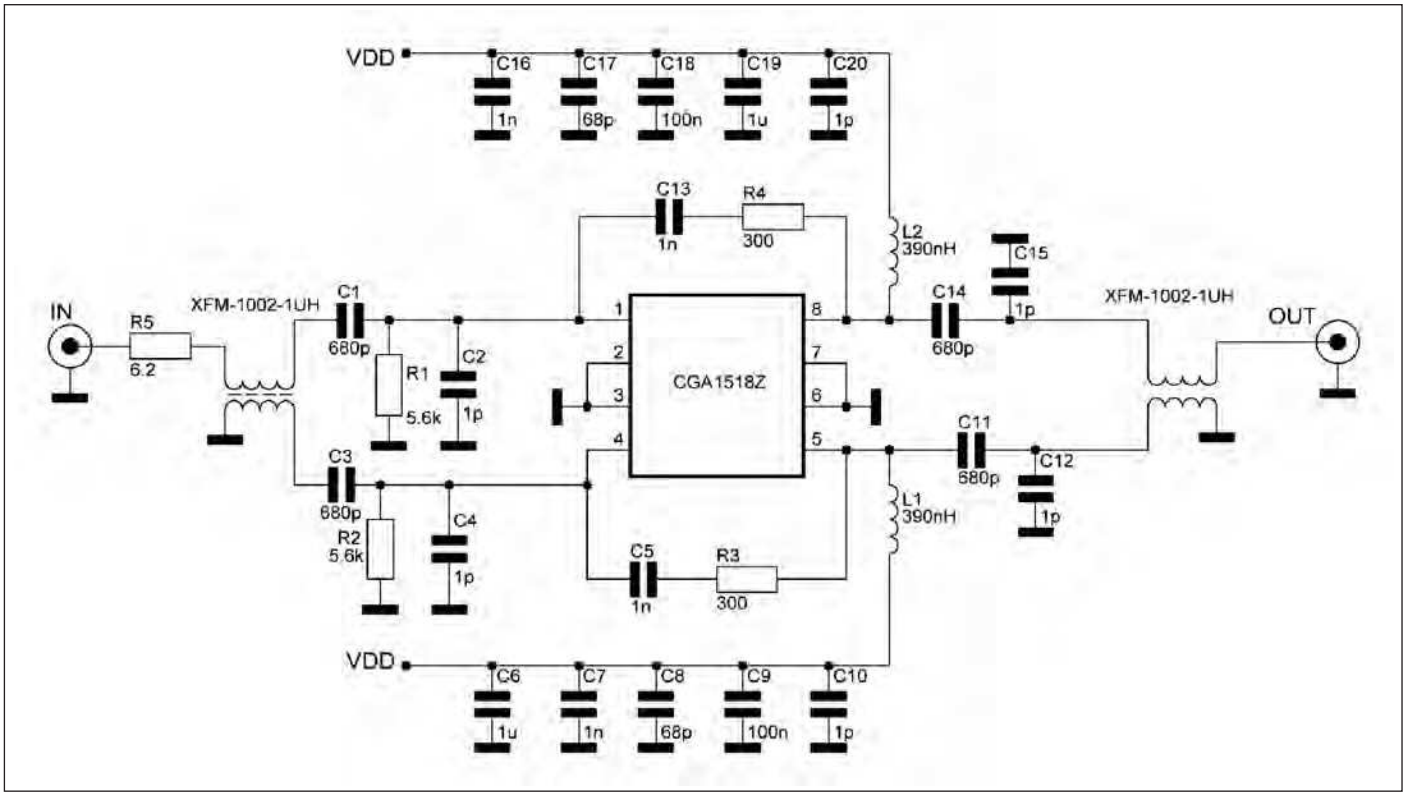
MICROCHIP

Il nome e il logo Microchip, il logo Microchip e PIC sono marchi registrati: mTouch è un marchio di Microchip Technology Incorporated negli U.S.A. e in altre nazioni. © 2010 Energizer. Energizer e altri marchi appartengono a Energizer. Tutti gli altri marchi qui menzionati appartengono ai rispettivi titolari. © 2010-2011, Microchip Technology Incorporated, Tutti i diritti riservati. ME257Blta/05.11

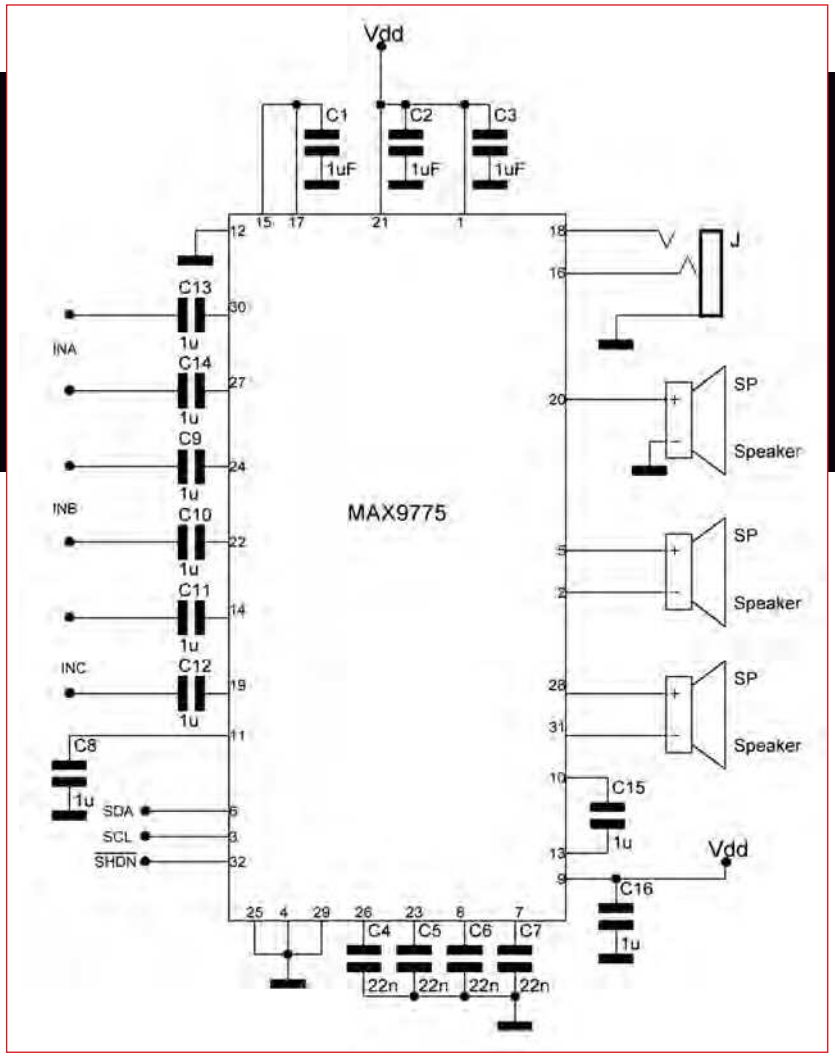


DRIVER SATA

L'STA1102R è un driver per dispositivi SATA in grado di pilotare trasferimenti fino a 3Gbps. Al suo interno ha due driver per pilotare due dispositivi SATA. È in grado di rilevare la presenza del segnale di ingresso, in modo tale da disattivare uno dei due driver per limitare i consumi. Opera con una tensione di 3.3V.



Amplificatore HBT *In figura è riportato lo schema di un amplificatore HBT, prodotto dalla RFMD. Questo integrato viene alimentato a 5V e presenta una configurazione Darligton per garantire la massima larghezza di banda del segnale.*

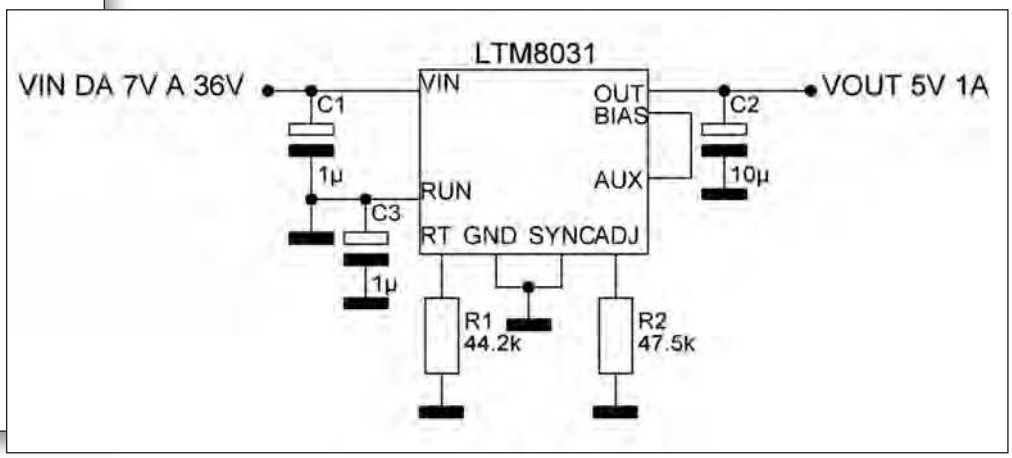


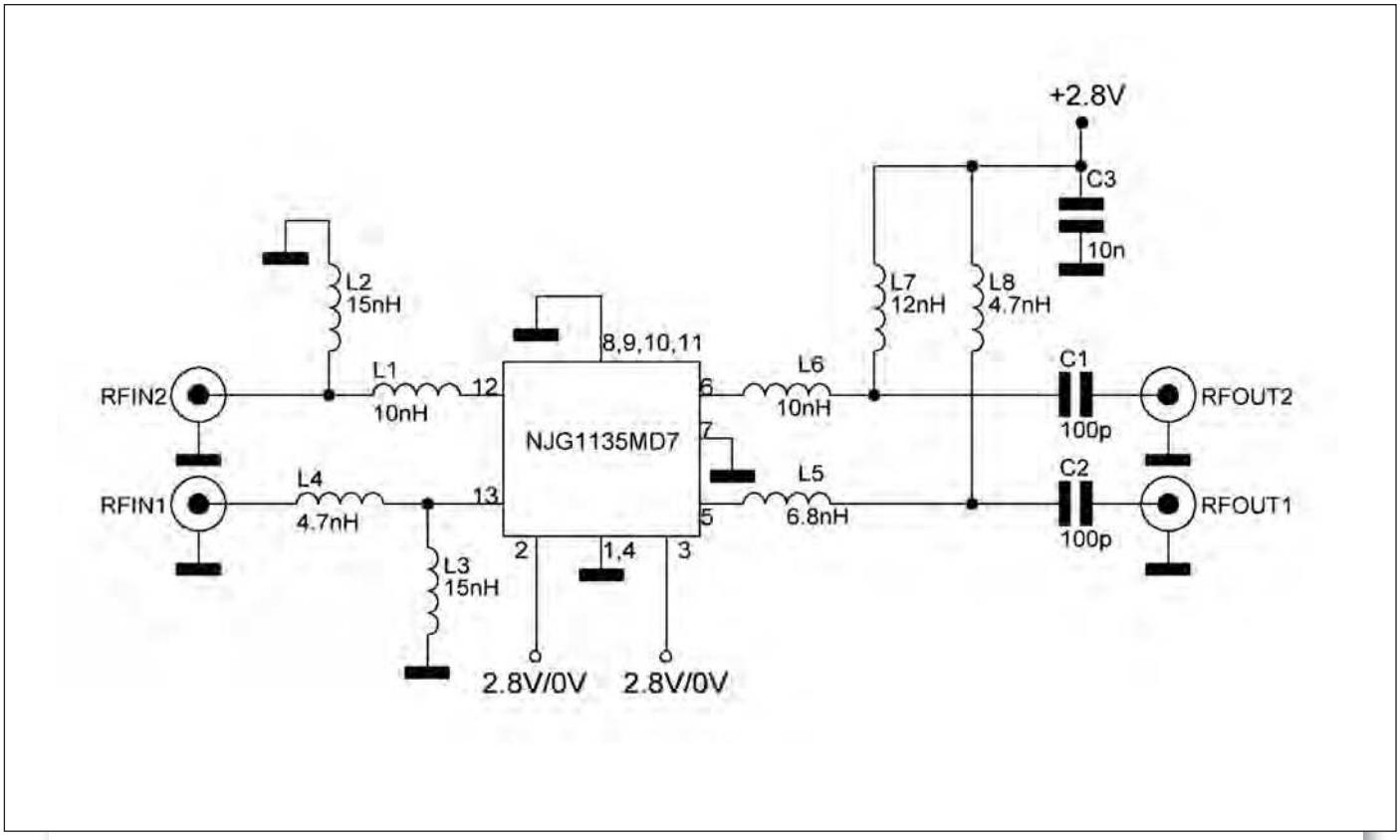
AMPLIFICATORE IN CLASSE D

Lo schema in figura è un amplificatore di classe D che, come altri amplificatori audio, può funzionare in modalità stereo e mono fornendo un watt e mezzo con una tensione di 5V su una impedenza di uscita di 4 ohm.

Regolatore da 1 A

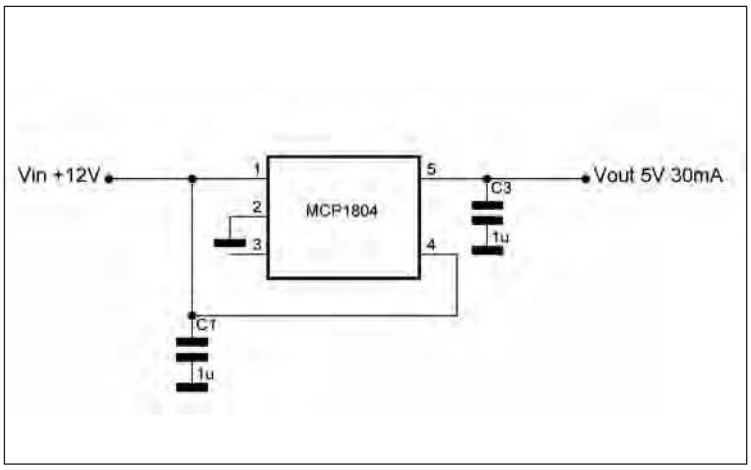
L'LTM8031 è un regolatore di tensione che fornisce una corrente costante di 1A. Accetta in ingresso un valore di tensione che va da 3.6V ai 36V, fornendo un valore di uscita da 0.8V a 10V. E' progettato secondo la normativa EN55022 per la minimizzazione dei disturbi dovuti ai campi magnetici dei vari componenti elettronici.





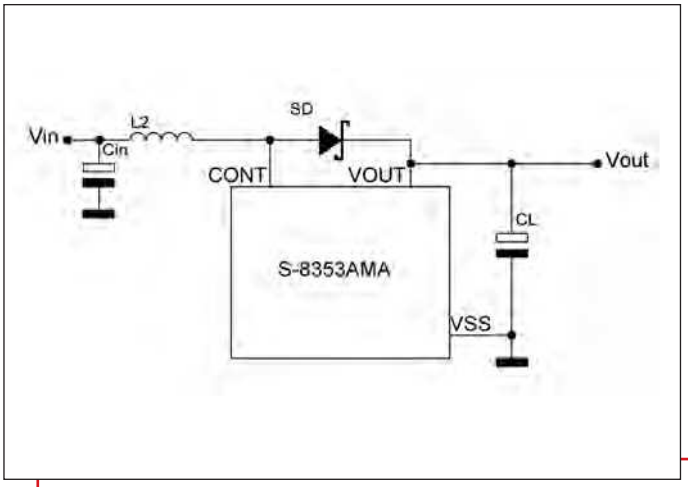
Guadagno regolabile

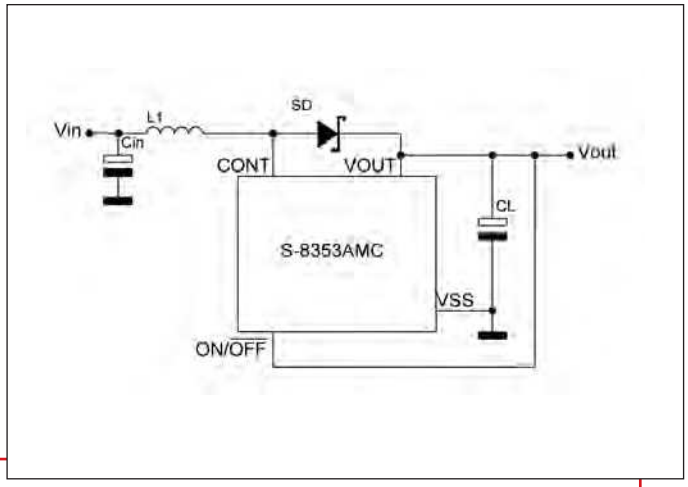
Il NJG1135MD7, su cui si basa lo schema di figura, è un LNA MMIC progettato per CDMA2000 a doppia banda per applicazioni per (cellulari e PCS). Implementa il comando per aumentare o diminuire il guadagno. Il sistema CMDA è un misto di tecnologia mobile tra la 2.5G e la 3G utilizzata nei telefonini per inviare contemporaneamente dati, voce e altri segnali.



REGOLATORE DA **150ma**

Nello schema è riportato un regolatore smd in grado di fornire una corrente massima di 150mA. Le tensione di ingresso vanno da 2V a 28V, mentre le tensioni di uscita sono comprese tra 1.8V e 12V. E' impiegato per dispositivi come videocamere, applicazioni per uso domestico e sistemi di navigazione.





ALIMENTATORE STEP-UP

I circuiti di figura costituiscono un alimentatore step-up con controllo PWM. Fornisce una tensione di uscita compresa da 1.5V a 6.5V ed è studiato per applicazioni portatili come PDA, PC portatili, cd players,ecc. Dispone inoltre di un controllo PFM, per evitare cali di tensione in corrispondenza di variazioni di carico. 🕮

Massimizzate la vostra velocità di collaudo da 200 MHz fino a 1 GHz

Il nuovo WaveSurfer MXs-B è progettato per mettervi a disposizione le massime prestazioni nei casi in cui la velocità di elaborazione e l'accuratezza sono decisive. La memoria e la frequenza di campionamento migliori della sua classe assieme all'ampia gamma di trigger e matematiche avanzate, unite allo strumento di ricerca ed analisi WaveScan™, consentono agli ingegneri di trovare glitch ed eventi rari più rapidamente.

- Banda da 200 MHz a 1 GHz
- 10 GS/s –Migliore nella sua classe
- 25 Mpts di memoria standard – Migliore nella sua classe
- Fino a 36 canali digitali – Migliore nella sua classe
- Ricerca ed analisi WaveScan™
- La gamma di opzioni per bus seriali più ampia disponibile
- Generazione di report e documentazione LabNotebook™

CODICE MIP 2804126

Nuovi oscilloscopi WaveSurfer MXs-B

Distributori regionali:

Vematron
Vematron S.r.l.
via Mons. Colombo, 34
21053 Castellanza (Va)
Tel. 0331 504064 | Fax 0331 505380
www.vematron.it | info@vematron.it

ema
E.M.A. S.r.l.
via del Maccabreccia 2 B-C,
40012 Calderara di Reno (Bo)
Tel 051 725441 | Fax 051 725406
www.ema.it | info@ema.it

a partire da 7140€

LeCroy
www.lecroy.com\it
Telefono 041-5997011

ANRITSU PREMIATO
DA FROST E SULLIVAN

Anritsu Company ha annunciato che i suoi analizzatori palmari di spettro MS272xC Spectrum Master hanno ottenuto il prestigioso premio New Product Innovation dell'anno 2011 da Frost e Sullivan, un'organizzazione leader nella ricerca di mercato a livello mondiale. Il premio è stato assegnato a Anritsu durante il banchetto Best Practices Awards 2011 di Frost & Sullivans, che riconosce le migliori fra le Aziende su scala mondiale. Nel selezionare MS272xC Spectrum Master come nuovo prodotto innovativo dell'anno, gli analisti e consulenti di Frost & Sullivan hanno valutato cinque criteri base che confrontano le performance dei principali analizzatori di spettro dai produttori di apparecchiature prova. Sono stati valutati in base agli elementi innovativi del prodotto, a come l'azienda sfrutti le tecnologie di punta, alle caratteristiche aggiunte e ai benefici, a come favorisca potenziali acquisizioni di clienti.

CODICE MIP 2807196

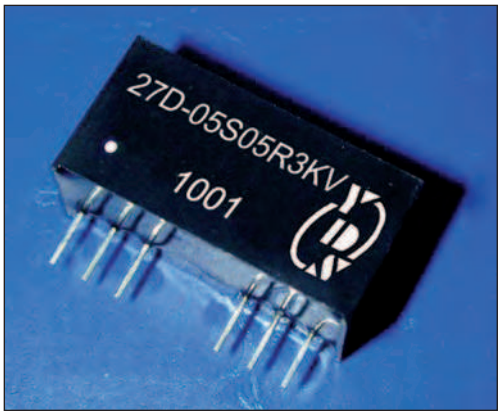
Come OMP MEchtron
garantisce la qualità

Per garantire la qualità dei propri prodotti, OMP Mechtron trova che sia fondamentale che test affidabili e accurati siano effettuati su circuiti stampati assemblati (PCBs su cui i connettori sono press-fitted) come garanzia di alta qualità per i clienti. Omp Mechtron S.p.a., una delle aziende europee leader nel campo della produzione meccanica, ricerca continuamente l'alta qualità e la soddisfazione dei propri clienti, per raggiungere una crescita economica e professionale. All'interno delle proprie strutture, conduce vari tipi di test elettrici (nel circuito, funzionali, isolamento elettrico, test dielettriche e produzione di test JIGs seguenti le specifiche dei clienti), così da fornire ai propri clienti prodotti sicuri, efficienti e senza alcun problema. Test elettronici effettuati per controllare la correttezza delle operazioni degli elementi del circuito e delle loro interconnessioni, sono vitali per assicurare operazioni di assemblaggio ottimali.

CODICE MIP 2807450

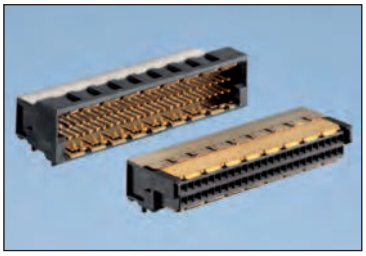
Convertitori DC/DC da 3000V di isolamento

MSC offre le serie 27D-3KV di moduli convertitori DC/DC 2Watt da YDS. Le principali caratteristiche di questi moduli convertitori DC/DC sono tensione di isolamento di 3000 VDC, protezione continua da cortocircuito, basso ripple e rumore, telecomando on/off, e una vasta gamma di temperature da -40°C a +85°C, con naturale raffreddamento a convenzione. I moduli convertitori DC/DC sono disponibili come modelli in uscita singoli o doppi e fornitori di tensioni di uscita unipolari o bipolaridi 5,9,12, 15 e 24 V con un'accuratezza di +/- 3%. Grazie alla precisa regolazione interna , la stabilità di ogni tensione in uscita è +/- 0.5% sui quattro intervalli di tensioni in entrata di 5-9 V, 9-18 V, 18-36 V e 36-72 V. Un MTBF di 1.5 milioni di ore per MIL-HDBK-217F a 25°C sottolinea l'alta qualità e affidabilità di questi moduli convertitori DC/DC, che sono forniti in singolo pacchetto 21.8 x 9.2 x 11.1 mm e sono compatibili con RoHS.



CODICE MIP 2807415

NUOVI CONNETTORI MICRO SPEED



Erni Electronics espande la sua famiglia di connettori Micro Speed di 1.0 mm includendo una versione a doppia fila a 90° in aggiunta ai connettori standard maschio e femmina, progettati per le tipiche applicazioni Mezzanine. I nuovi connettori MicroSpeed supportano una velocità di trasmissione dati fino a 10 Gbit/s e sono i risultati di un lavoro intenso di sviluppo che utilizza nuove tecnologie di connessione e nuovo materiale, combinati ai contatti di Erni e alla completa schermatura dei lati superiore e inferiore. Il 90° MicroSpeed ha stabilito nuovi standard di performance per velocità di trasferimento dati (fino a 10 Gbit/s), dimensione e robustezza, e permette connessioni ultra compatte sia da backplane a schede derivate, sia per connessioni complanari tra scheda e scheda.

CODICE MIP 2807188

Mentor Graphics introduce la nuova piattaforma Calibre RealTime

Mentor Graphics Corporation ha presentato la nuova piattaforma Calibre RealTime per la physical verification durante la creazione del progetto a qualità garantita. La prima release fornisce contestualmente design rule checking (DRC) e DFM istantanei nella soluzione SpringSoft Laker custom IC. La nuova soluzione implementa velocità e qualità dei risultati, dando ai progettisti, per la prima volta, tutto il potenziale dei signoff engines e qualified decks di Calibre durante il disegno. Questo permette ai progettisti di ottimizzare le loro performance di lay-out senza sacrificare il manufacturing. Una versione per ambiente IC Station di Mentor sarà disponibile a Giugno.

CODICE MIP 2807684

CAMPAGNA ABBONAMENTI

PROMO tutte 3

RISPARMIO ASSICURATO!
Risparmierai fino a 73,00 EUR sul prezzo di copertina. Garanzia del prezzo bloccato per un anno. Le riviste ti saranno recapitate comodamente a casa. Riceverai un buono sconto del 20% per un tuo prossimo acquisto su www.ieshop.it

UNA PROMOZIONE ESCLUSIVA:

- 11 numeri di FARE ELETTRONICA
- 11 numeri di ELEKTOR
- 11 numeri di FIRMWARE (edizione digitale) compreso l'accesso all'archivio a partire dal numero 49 di febbraio 2010
- 1 anno di accesso al CLUB di FARE ELETTRONICA
- 1 anno di accesso al CLUB di FIRMWARE,
- 20% di sconto per un acquisto su www.ieshop.it

TUTTO A SOLI € 119,50 ANZICHE' € 192,50

Sì, mi abbono alla PROMOTUTTE3 (è possibile abbonarsi online www.ieshop.it/promotutte3)

Riceverò 11 numeri di Fare Elettronica e 11 di Elektor in edizione cartacea, potrò accedere all'archivio di Firmware in edizione digitale a partire dal numero 49 di febbraio 2010 e ai prossimi 11 numeri, compreso un anno di accesso al CLUB di Fare elettronica e Firmware, tutto a **soli € 119,50** anziché € 192,50

METODI DI PAGAMENTO

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **ALLEGO ASSEGNO** (Intestato a Inware Edizioni)

☐ MasterCard ☐ VISA ☐ postepay

PayPal fare il pagamento su paypal@inwaredizioni.it

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, via Giotto 7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . .
Cap . . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . .
Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

1 PERCHÉ SCEGLIERE FARE ELETTRONICA:

LAVORO: Se lavori come progettista elettronico, Fare Elettronica saprà esserti utilissima con numerose idee e spunti sia dal punto di vista pratico che teorico. DIDATTICA: se lavori come insegnante in una scuola ad indirizzo tecnico, Fare Elettronica risulterà tra le migliori fonti di ispirazione per te e per i tuoi allievi. AGGIORNAMENTO TECNOLOGICO: se sei alla ricerca di una fonte d'ispirazione, Fare Elettronica non ti deluderà, fornendoti l'aggiornamento sulle più recenti tecnologie, trattate in modo esauriente e competente. HOBBY: se invece l'elettronica è la tua passione, Fare Elettronica ti permetterà di coltivarla giorno per giorno con progetti sempre appassionanti ed alla tua portata.

2 PERCHÉ SCEGLIERE FIRMWARE:

Firmware è una fantastica rivista digitale, tra le prime uscite nel settore dell'elettronica e la prima in assoluto in lingua italiana. Nasce come naturale evoluzione della rivista Firmware in edizione cartacea, rivista dedicata ai professionisti dell'elettronica, esigenti e competenti bisognosi di uno strumento di aggiornamento e di know how per l'elettronica professionale. Firmware è basata su una piattaforma che saprà regalarti una nuova esperienza di accesso a incredibili contenuti. Nessuna rivista ti sembrerà più all'altezza dopo aver visto Firmware!

3 PERCHÉ SCEGLIERE ELEKTOR:

Elektor è in circolazione in oltre 50 Paesi con un totale di 300,000 copie mensili. Pubblicato in Inghilterra, Francia, Paesi Bassi, Spagna, Grecia, Portogallo, Brasile, Svezia, Finlandia, Germania, Italia, Stati Uniti. La rivista Elektor è distribuita a livello mondiale tramite abbonamento, edicole e librerie. I siti internet sono tradotti in Inglese, Tedesco, Francese, Italiano, Cinese, Portoghese, Brasiliano, Svedese e Greco. 2 milioni di lettori mensili. 140.000 visitatori web mensili. 43.000 abbonati a e-weekly, la newsletter digitale. Elektor propone progetti di elevata qualità ideali per l'autocostruzione nei campi dei microcontrollori, audio high-end, RC modelling, RF e test & measurement. Elektor è creata da veri appassionati di elettronica. Elektor è l'unica rivista internazionale indipendente a vantare un laboratorio di progettazione e PCB design interno.

PERCHÉ SCEGLIERE TUTTE E TRE LE RIVISTE

Perché solo Inware Edizioni può darti una offerta così completa a condizioni così convenienti con un mix di aggiornamento, conoscenza, stimolo, divertimento e professionalità! Approfitta della fantastica PROMOTUTTE3 non te ne pentirai

LA RIVISTA DI ELETTRONICA PIÙ LETTA AL MONDO (OLTRE 400MILA COPIE IN 12 LINGUE) DA OGGI IN ITALIANO!

*luglio/agosto numero doppio

Creata da veri appassionati di elettronica con la produzione di molti progetti pratici, tutti originali e sempre innovativi.

Progetti di elevata qualità ideali per l'autocostruzione nei campi dei microcontrollori, audio high-end, RC modelling, RF e test & measurement.

Gli sviluppi sulle nuove tecnologie, grazie ad una serie di articoli mensili, sempre pratici e visti da angolazioni diverse.

L'unica rivista al mondo che può vantare laboratori di progettazione e di disegno dei PCB, con la produzione e vendita di kit.

Il nuovo numero ti aspetta in edicola a partire dal 20 del mese precedente*

www.elektor.it

CODICE MIP 2808425

11-12 giugno 2011
GRANDE FIERA dell'Elettronica

Fiera dell'elettronica, radiantistica, surplus, telefonia. Grandi occasioni anche per informatica, CD, DVD e stampa specializzata.
Dove: Longarone (BL) **Quando**: 11-12 Giugno 2011
Orari: dalle 9.00 alle 18.00 **Organizzazione:** Compendio Fiere
info:www.compendiofiere.it

CODICE MIP 2806572

24-26 giugno 2011
MOSTRA DELL'ELETTRONICA

Computer e telefonia, informatica, elettronica, antenne e TV/SAT, Hi-Fi, sicurezza informatica, editoria specializzata. Queste le tematiche della terza edizione della Mostra dell'elettronica in Valle d'Itria.
Dove: Martina Franca (TA) **Quando:** 24-26 Giugno 2011
Orari: dalle 9.00 alle 18.00 **Organizzazione:** Anse Fiere
info: www.ansefiere.it

CODICE MIP 2806574

1-3 luglio 2011
FIERA DELL'ELETTRONICA di PUGLIA

Fiera dell'elettronica, radiantistica, surplus, telefonia. Grandi occasioni anche per informatica, CD, DVD e stampa specializzata. Mercatino della radio d'epoca, grande appuntamento per i radioamatori.
Dove: *Gravina di Puglia (BA)* Quando: *1-3 Luglio 2011*
Orari: *dalle 9.00 alle 18.00* **Organizzazione:** *Anse Fiere*
info:www.ansefiere.it

CODICE MIP 2808282

2-3 luglio 2011
GRANDE FIERA DELL'ELETTRONICA

Fiera dell'elettronica, radiantistica, surplus, telefonia. Grandi occasioni anche per informatica, CD, DVD e stampa specializzata.
Dove: Monsummano Terme (PT) **Quando**: 2-3 Luglio 2011
Orari: dalle 9.00 alle 18.00
Organizzazione: Compendio Fiere
info:www.compendiofiere.it

CODICE MIP 2808332

24-26 GIUGNO 2011 HAM RADIO

Un appuntamento storico per i radioamatori. L'Ham Radio di Friedrichshafen è la fiera europea di riferimento per radioamatori ed hobbisti elettronici che attrae visitatori da tutto il mondo. In occasione della fiera si terranno workshop e conferenze di sicuro interesse per gli appassionati del settore.
Dove: Friedrichshafen (Germania)
Quando: 24-26 Giugno 2011
Orari: dalle 9.00 alle 18.00
Organizzazione: Messe Friedrichshafen
info: www.hamradio-friedrichshafen.de

CODICE MIP 2806251

di MAURIZIO DI PAOLO EMILIO

Com'è fatto UN PC?

La scheda madre è il circuito elettronico più importante in un PC. Viene appunto chiamata "madre" (motherboard o mainboard) perché tutto, all'interno del PC, vi è collegato e tutto deve comunicare con essa affinché il PC possa funzionare. Analizziamola in dettaglio insieme ai suoi componenti principali

La scheda madre **(figura 1**) è la base su cui poggia la struttura del PC. È possibile paragonarla alle fondamenta che sostengono un palazzo, determinandone la costruzione. Migliori sono le fondamenta e più stabile e affidabile sarà il palazzo. La scheda viene montata all'interno del computer e su di essa è alloggiato ogni singolo componente hardware del PC, fra i quali il **processore**, la memoria **RAM**, il **BIOS** e le varie schede di espansione (**scheda video**, **scheda audio**, **modem**, **scheda di rete**...). Sono inoltre presenti i connettori per le memorie di massa (come **Hard Disk**, **Floppy Disk** e **CD-ROM**) e le porte di comunicazione, cioè i connettori a cui vengono collegate le periferiche esterne (come **tastiera**, **mouse**, **monitor**, **stampante, ecc**...). Ciascuno di questi componenti è in grado di comunicare con gli altri proprio grazie alle connessioni elettriche presenti sulla scheda madre. La scheda madre, infatti, ha il compito di gestire le entrate e le uscite del computer e di regolare il flusso dei dati tra i componenti. In altre parole, la scheda madre controlla tutto, dall'arrivo dei segnali dai tasti della tastiera, all'invio di questi segnali al processore, dal ricevimento dei risultati dell'elaborazione, al loro passaggio all'interfaccia video per il monitor. La scheda madre poi svolge altri compiti, come salvare i dati sull'hard disk o ricevere suoni da una sorgente esterna.

I COMPONENTI: LA CPU E IL BIOS

L'Unita Centrale di Elaborazione è comunemente indicata con l'acronimo CPU (Central Processing Unit). Il processore è di fatto il "cervello" del computer. Il processore recupera dalla memoria centrale le istruzioni da eseguire e al termine dell'elaborazione dei calcoli, il risultato è restituito nuovamente alla memoria centrale. La velocità con la quale queste operazioni sono eseguite è determinata dalla frequenza di lavoro del processore, dalla modalità con cui vengono eseguite le operazioni volte al secondo, dalla quantità e velocità della memoria centrale con cui la CPU scambia le informazioni.
La sigla BIOS sta per Basic Input/Output System e costituisce il software di base funzionante su un PC e viene installato sul PC direttamente in fabbrica. Il BIOS contiene tutte le istruzioni necessarie affinché la CPU possa comunicare con la tastiera, con il mouse, il monitor, il disco fisso e i dispositivi di comunicazione all'avvio del computer, il BIOS si oc-

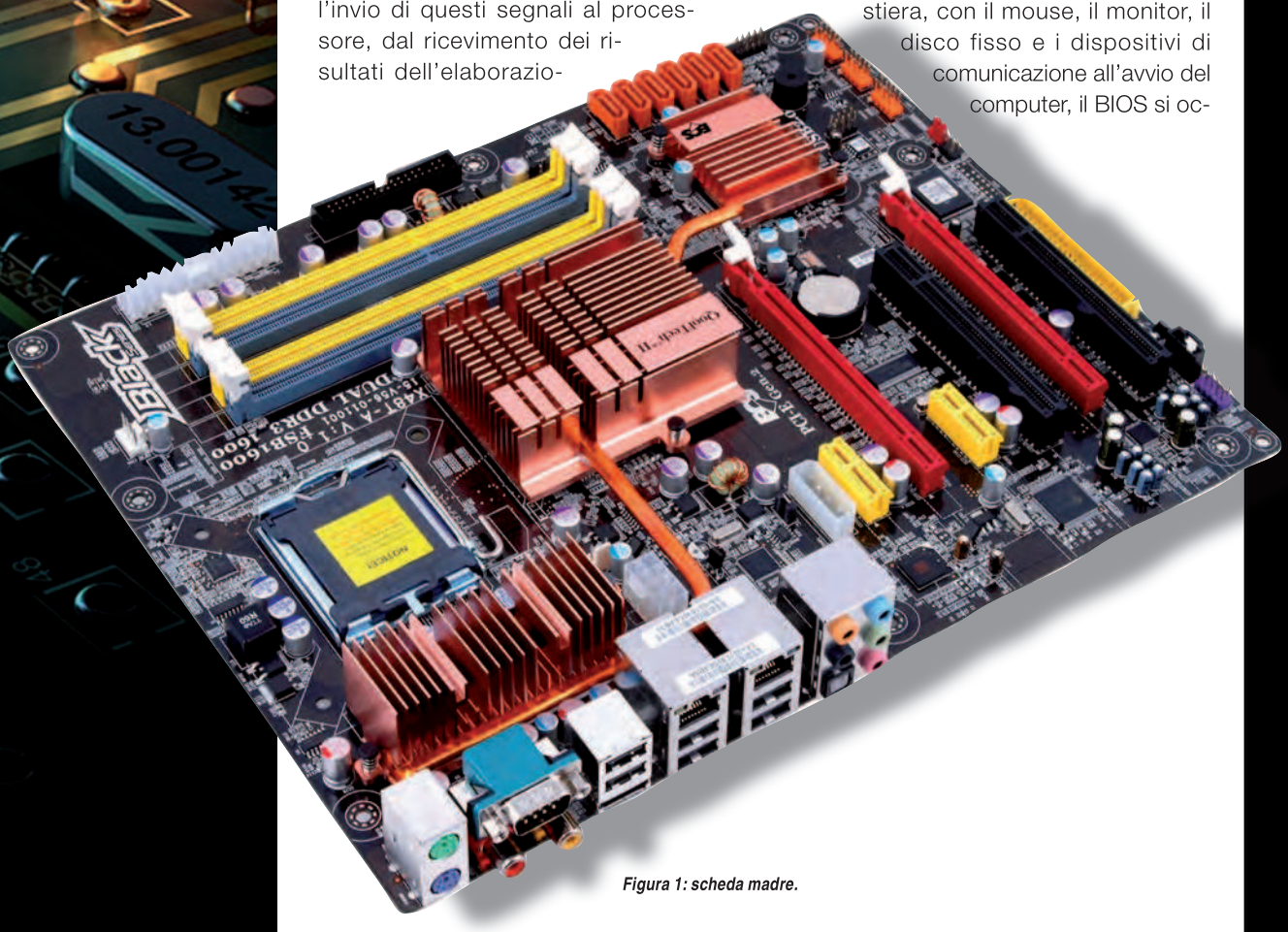


Figura 1: scheda madre.

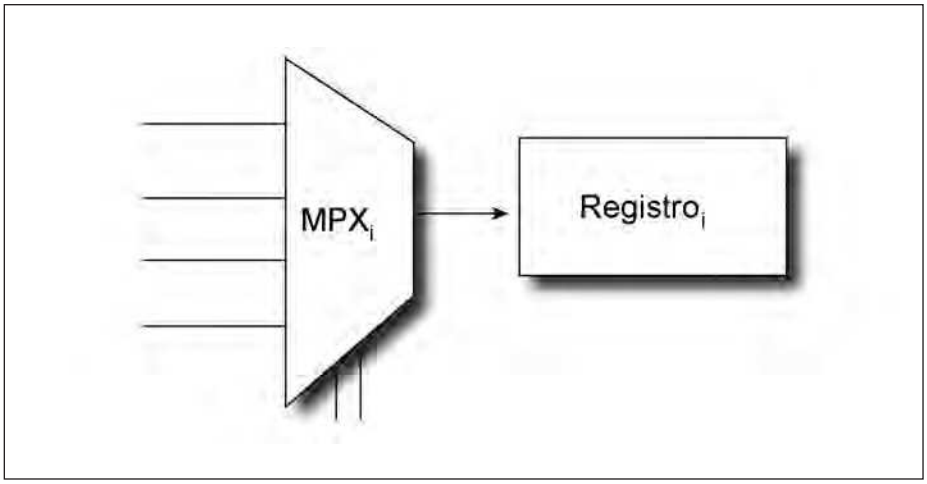


Figura 2: CPU, possibile schema.

cupa di caricare, da uno dei dischi, il sistema operativo (Windows, Linux, Leopard ecc...) e svolge funzioni di configurazione e controllo delle varie componenti, fino a portare il PC nelle condizioni di essere pienamente operativo. Poiché il BIOS è di fatto un programma, normalmente memorizzato su una ROM (Read Only Memory) riscrivibile di tipo Flash, è possibile aggiornarlo ("flashare il BIOS") tramite appositi programmi reperibili sul sito Web del produttore della scheda madre. La CPU è una "macchina" di scopo generale: sa fare un certo numero di operazioni e, tramite queste, è in grado di risolvere pressoché ogni problema. Le istruzioni che la CPU deve eseguire per risolvere un problema, in quanto macchina generica, devono essere assunte dall'esterno, in particolare dalla memoria RAM mediante una sequenza nota come Fetch. L'istruzione prelevata viene inserita nell'Instruction Register (IR) e qui è pronta per essere eseguita. I vari componenti della CPU (ALU, registri, CU,...) devono, per poter operare, comunicare tra loro. Un modo per realizzare tale comunicazione potrebbe essere quello di **figura 2**. Qui le linee di ingresso del multiplexer arrivano dalle uscite dei registri e le linee di selezione arrivano dalla CU. In questo modo si ha la possibilità di fare, ad esempio, due trasferimenti contemporanei da due registri a due destinazioni arbitrarie. Un modo alternativo, meno costoso dal punto di vista del numero di componenti, è quello di **figura 3**.

Vediamo ora in dettaglio la struttura a singolo bus interno di una CPU teorica **(figura 4)**:

- ALU: serve per fare somme, sottrazioni ed operazioni logiche.
- Y contiene uno dei due operandi necessari per poter svolgere un'operazione con la ALU. Tale registro è richiesto per il fatto che i collegamenti si basano tutti su un singolo bus. L'altro operando viene prelevato dal bus.
- MDR (Memory Data Register) è un registro che contiene dati e che comunica con l'esterno con il bus dati.
- IR (Instruction Register). È un registro che contiene l'istruzione che è correntemente in esecuzione. È qui che viene memorizzata l'istruzione non appena è stata compiuta la fase di Fetch. Ad ogni istruzione corrisponde un numero univoco, la codifica dell'istruzione in macchina, che la rappresenta.

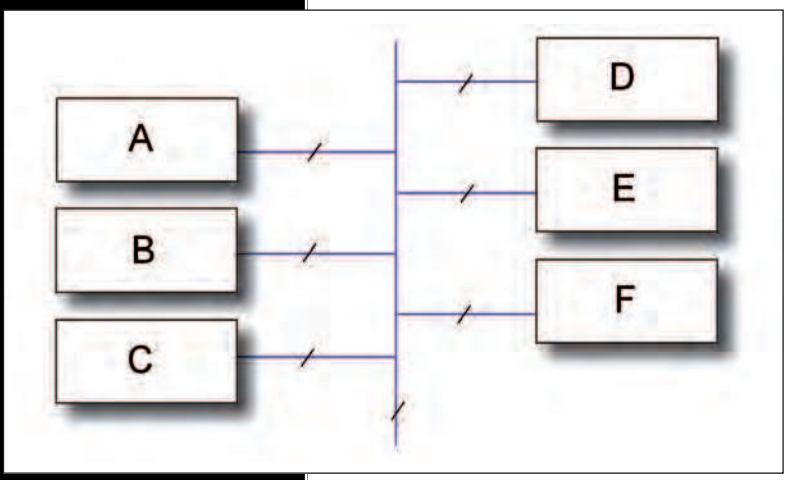
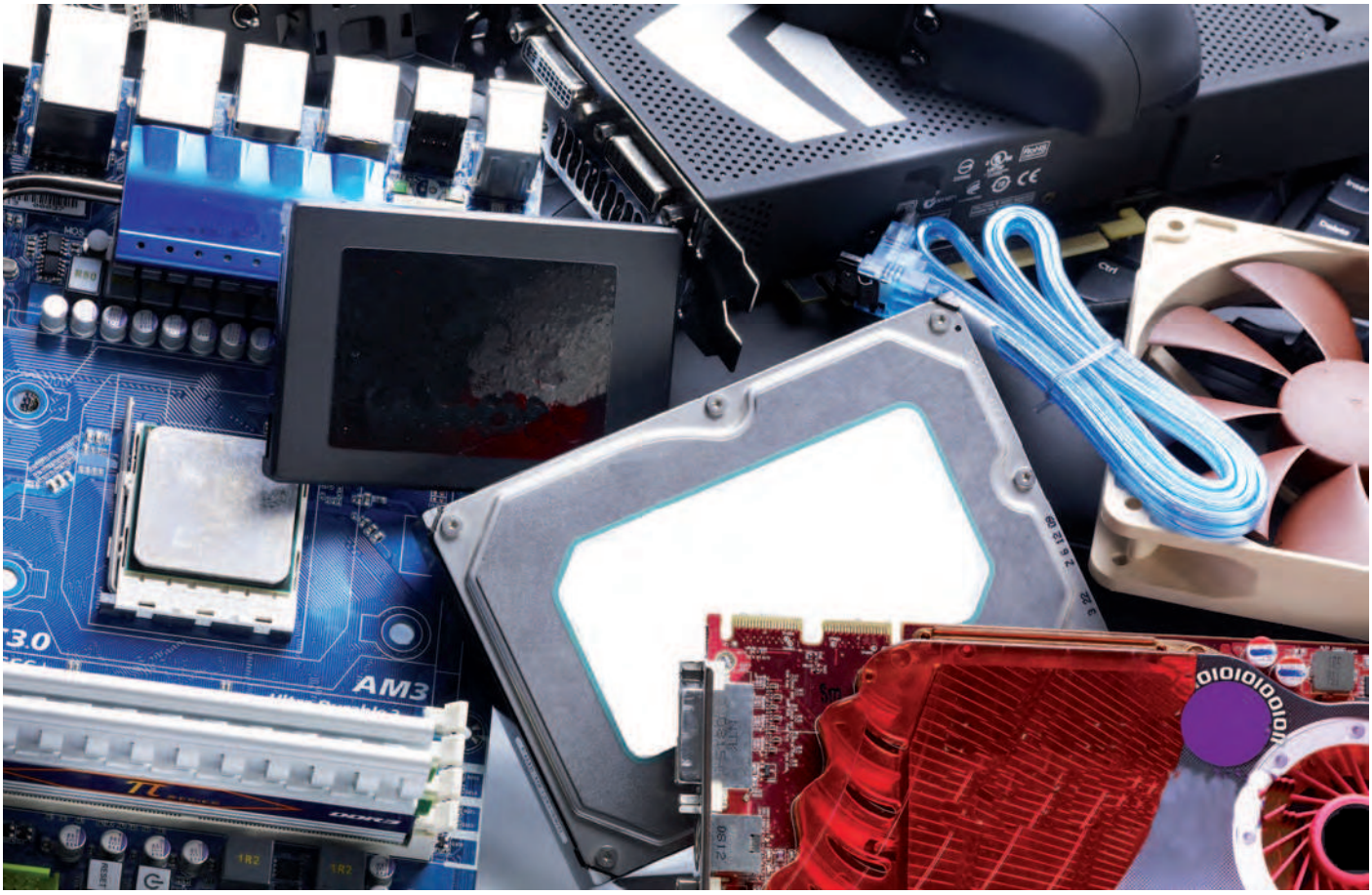


Figura 3: CPU, un altro possibile schema.



• PC (Program Counter): è un registro il cui contenuto è l'indirizzo dell'istruzione successiva all'istruzione corrente.

Raffreddamento CPU

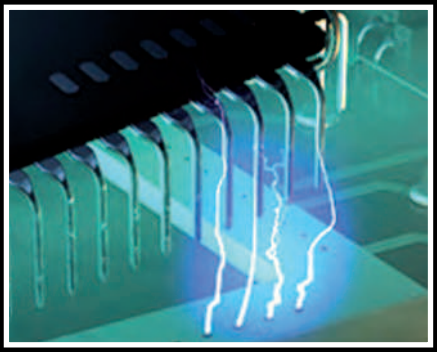
La CPU è notoriamente il componente che dissipa la maggior quantità di calore; a titolo di esempio, i più recenti microprocessori disperdono, sotto forma di calore, ben il 30% dell'energia che ricevono. Sebbene i produttori stiano lavorando per ridurre questo valore, riducendo il canale dei transistor e agendo sul tipo di metallizzazione e stratificazione del silicio, oggigiorno i processori richiedono sistemi decisamente più complessi di un tempo per raffreddare e rendere loro affidabili.

Se avete scoperto che la vostra CPU lavora a temperature non idonee, è tempo di effettuare un intervento dedicato. Nella maggior parte dei casi, è possibile risolvere il problema senza cambiare dissipatore, controllando quanto segue: la disposizione della pasta termoconduttiva; posta tra dissipatore e CPU, migliora il contatto tra di essi, aumentando la capacità del dissipatore stesso. La corretta rotazione delle ventole; non sono eterne, possono rompersi e perdere l'allineamento o semplicemente essere appensantite dalla polvere.

Ogni dissipatore accoppiato ad un tipo di socket, necessita di una particolare installazione, bisogna, dunque, riferirsi al manuale del dissipatore e conoscere bene il tipo di scheda madre su cui stiamo agendo. Se queste misure dovessero rivelarsi insufficienti, è necessario acquistare un dissipatore più potente o perlomeno sostituire la ventola del dissipatore che già si possiede con un modello dotato di maggior CFM (Cubic Feet per Minute, piedi cubici al minuto), cioè maggior quantità di aria spostata nel'unità di tempo. Qui entriamo in un campo molto vasto, su cui talvolta pure le certezze scarseggiano. Non è nostro compito entrare nel merito dei vari modelli di dissipatore, ognuno va scelto in base a tanti fattori, potenza da smaltire, tipo di socket, tipo di scheda madre, tipo di case, ecc.

I COMPONENTI: LE MEMORIE

La RAM (*Random Access Memory, memoria ad accesso casuale*), è la memoria centrale del calcolatore e lavora a stretto contatto con il processore, che attinge da essa le istruzioni da eseguire e i dati da elaborare. A differenza delle cosiddette memorie di massa, che mantengono le informazioni anche a computer spento, la RAM è una memoria di tipo volatile, cioè perde il suo contenuto quando non viene più alimentata (viene spento il computer). La quantità di memoria installata in un computer ne influenza notevolmente le prestazioni: se, infatti, la RAM non è sufficiente a contenere tutti i programmi in esecuzione, il sistema operativo utilizza il disco rigido per estenderla "*virtualmente*" (meccanismo della memoria virtuale, noto anche come *swapping*). Il meccanismo è utilissimo perché altrimenti il computer non sarebbe in grado di eseguire programmi con dati molto voluminosi (ad esempio, certi programmi di manipolazione grafica). Poiché però i dischi rigidi hanno tempi d'accesso nell'ordine dei



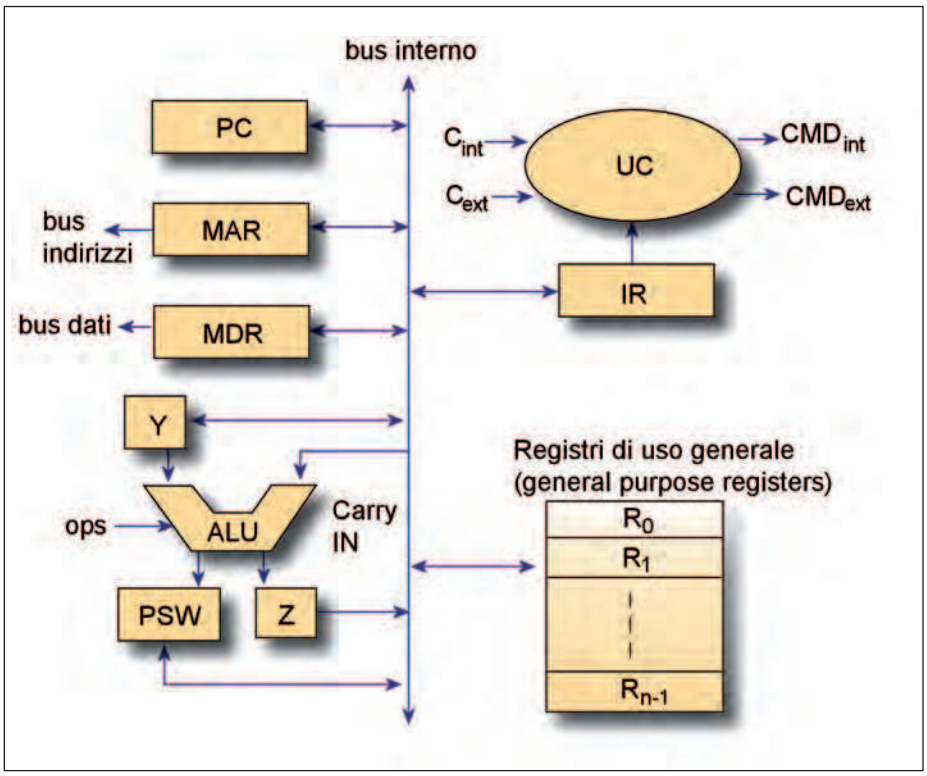


Figura 4: schema a blocchi CPU, singolo bus interno.

millisecondi (contro i *nanosecondi* delle memorie RAM), l'utilizzo dell'hard disk come memoria virtuale provoca un notevole peggioramento delle prestazioni. Con i sistemi operativi attuali è consigliabile disporre di almeno 2 GB di RAM, in modo da limitare lo swapping a casi eccezionali. La regola pratica, comunque, è che quando il vostro hard disk entra in funzione troppo di frequente è tempo di espandere la RAM.

Il disco fisso HDD (*Hard Disk Driver*) viene inserito in uno degli scomparti disponibili nel PC Case e viene fissato mediante delle viti apposite. Viene alimentato da un cavo a 4 conduttori proveniente dall'alimentatore. I dati da e verso la scheda madre vengono trasportati mediante un cavo IDE a 40-pin o, per i dischi di recente generazione, un cavo SATA. I dati vengono immagazzinati magneticamente su più dischi fissi interni disposti a pila uno sull'altro. Piccoli bracci con testine magnetiche si muovono rapidamente avanti e indietro su entrambe le superfici di ciascun disco. I sensori si trovano a pochi micron sopra la superficie del disco rotante e sono in grado di leggere e scrivere dati a velocità molto elevate.

I COMPONENTI: L'ALIMENTAZIONE

L'alimentatore è un trasformatore che converte la corrente elettrica di linea in corrente continua a 5 e 12 volt, necessaria ai componenti del PC. Esso è installato in un angolo del PC Case, vicino alla scheda madre. Un cavo conduttore trasporta corrente continua da +5V, -5V e da +12V, -12V alla scheda madre.

Un'altra coppia di cavi, ciascuna con quattro conduttori e due connettori a 4-pin avvolti tra loro, trasportano corrente continua a +5v e +12v ai dischi (Hard Disk HDD, CD, DVD). Inoltre, l'alimentatore contiene un ventilatore di raffreddamento che aspira aria dall'interno del PC per tenere i componenti essenziali, come il processore e il trasformatore, alla corretta temperatura di funzionamento.

È questo che genera il sibilo che si sente quando il PC è acceso. I moderni PC hanno ventole di raffreddamento anche sul processore, sulla scheda video, sull'Hard Disk HDD.

Un alimentatore tipico in un PC ha una potenza di 300-350 watt. Sono possibili anche potenze superiori. ❑

CODICE MIP 2804859

DSO Nano V2

Raddoppia la banda passante

2798973 2797234

Oscilloscopio tascabile digitale

Seconda versione dell'oscilloscopio DSO NANO, caratterizzata da una più ampia banda passante (1MHz) e nuovo design con finiture in metallo. Lo strumento visualizza la forma d'onda della tensione d'ingresso su un display LCD da 2,8" a colori, memorizzando all'occorrenza i tracciati su una Micro SD-Card. Integra un generatore di segnale ad onda quadra e dispone di una presa mini-USB che permette di scaricare e visualizzare su un PC le forme d'onda e di aggiornare il firmware.

***DSONANOV2* — € 118,00**

ACCESSORI:

Cavetto adattatore da jack 3,5mm a BNC
DSOJACKBNC — € 2,60

Sonda di ricambio a due puntali
DSOSONDA — € 2,00

Prezzi IVA inclusa

FUTURA ELETTRONICA

Via Adige, 11 - 21013 Gallarate (VA)
Tel. 0331/799775 ~ Fax. 0331/792287

Disponibili presso i migliori negozi di elettronica o nel nostro punto vendita di Gallarate (VA)
Caratteristiche tecniche e vendita on-line direttamente sul sito www.futurashop.it

GESTIONE DI *elettropompe* con ARDUINO

Una curiosa ed utile applicazione di Arduino per assicurare i vostri locali da possibili allagamenti...

Uno dei maggiori problemi che affligge chi ha locali posti sotto al livello stradale (garage e/o cantine), è quello dei possibili allagamenti: spesso è sufficiente un acquazzone estivo per riempire i pozzetti di raccolta dell'acqua, causando così l'allagamento dei locali in questione, con tutti i danni che ne derivano. Generalmente, il problema si risolve tramite l'uso di pompe di sollevamento. Le soluzioni commerciali hanno il difetto di essere particolarmente costose (elettropompe abbinate ad un inverter ed al gruppo soccorritore) oppure di non essere particolarmente efficaci (una pompa alimentata dalla tensione di rete e comandata da un galleggiante: un semplice black out la rende totalmente inutile). Ecco quindi che l'utilizzo di pompe funzionanti in bassa tensione (12V) e alimentate da batterie risolvono il problema. Se, invece, di impiegare un'unica pompa se ne impiegano due, il sistema diventa decisamente più affidabile ed efficace. Se, infine, comandiamo il tutto tramite Arduino, la cosa diventa pure divertente da realizzare e anche economica (particolare da non disprezzare). La soluzione proposta prevede l'utilizzo di due pompe che vengono fatte funzionare in maniera alternata: ogni volta che il sistema entra in funzione, utilizza la pompa che precedentemente non era stata utilizzata. In questa maniera si ha la sicurezza che ciclicamente entrambe le pompe vengono fatte funzionare. Se si avesse una pompa principale e una di riserva, si cor-

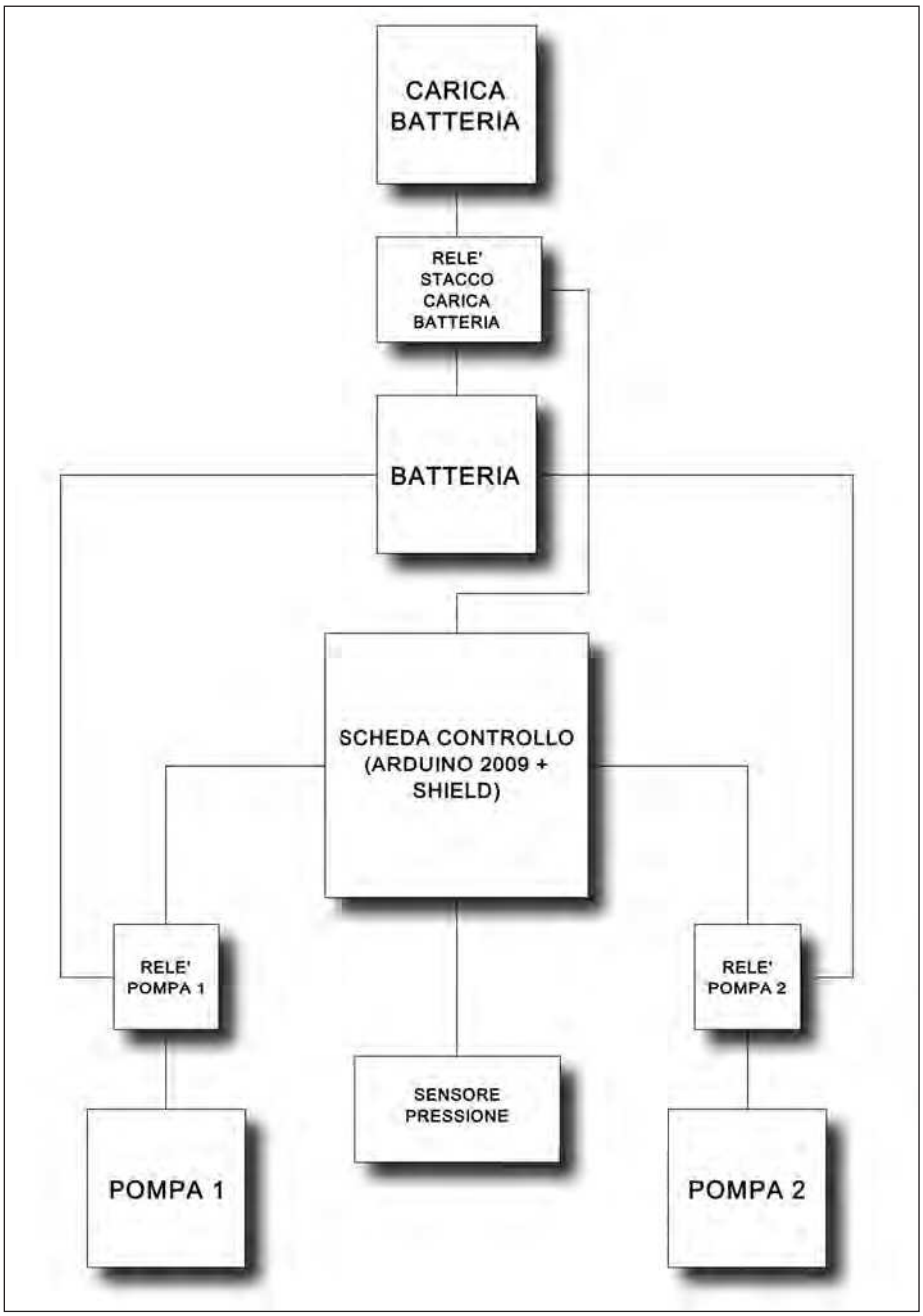
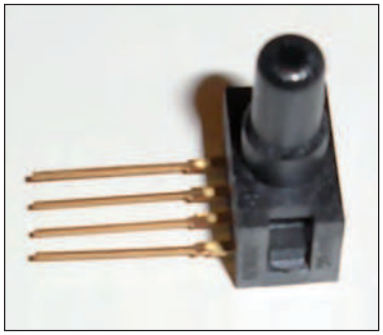


Figura 1: schema a blocchi del sistema.

rerebbe il rischio di tenere ferma per troppo tempo la pompa di riserva con il conseguente rischio di ritrovarsi con una pompa non funzionante proprio nel momento del bisogno. Se il sistema rileva che il livello dell'acqua ha oltrepassato un certo limite, attiva, contemporaneamente, entrambe le pompe per avere una maggiore capacità di estrazione dell'acqua. In **figura 1** è rappresentato lo schema a blocchi del sistema. Cuore del sistema è la scheda "Arduino 2009" che provvede a tenere sotto controllo il livello dell'acqua tramite un apposito sensore e a comandare (in maniera alternata oppure contemporaneamente, in base alle necessità) le due pompe. Come pompe si è pensato di utilizzare delle comuni pompe di sentina (normalmente utilizzate nel settore nautico). Tale tipologia di pompe presenta il vantaggio di funzionare a 12V (quindi, è possibile alimentarle tramite una normale batteria al piombo da automobile), di essere economiche e facilmente reperibili. In un negozio di nautica si possono trovare vari modelli di pompe di sentina con differenti portate: la scelta andrà quindi fatta in base alla quantità di acqua da estrarre. In **figura 2** si può vedere una tipica pompa di sentina utilizzata nel settore della nautica. Come sensore di livello si è deciso di impiegare un sensore di pressione collegato a un tubetto in gomma (che andrà immerso nell'acqua). Il principio di funzionamento è il seguente: se il livello dell'acqua aumenta, l'aria contenuta all'interno del tubetto in gomma verrà compressa, determinando un aumento di pressione che verrà, quindi, rilevato dal sensore stesso. La pressione misurata sarà, quindi, proporzionale al livello dell'acqua. Tale tipo di sensore evita l'utilizzo di galleggianti che risultano essere sicuramente più ingombranti rispetto ad un tubetto del diametro di qualche millimetro.

Figura 2: pompa di sentina utilizzata nel settore nautico.

Figura 3: sensore di pressione utilizzato come sensore di livello



Il sensore di pressione, disponendo di un'uscita analogica, ci permetterà di misurare il livello dell'acqua (e non solo di sapere se il livello ha superato un certo limite) consentendoci, così, di decidere se azionare una sola pompa oppure entrambe le pompe contemporaneamente. Come sensore di pressione è stato utilizzato il modello 24PCAFA6G della Honeywell. In **figura 3** è possibile vedere il sensore utilizzato nella nostra applicazione. In figura 4 è rappresentato lo schema di montaggio del sensore. Tutto il circuito è alimentato tramite una batteria al piombo del tipo usato nelle automobili. La batteria viene tenuta sotto carica da un semplice caricabatterie (reperibile in qualsiasi negozio di ricambi per auto).

Quando la scheda Arduino attiva una delle due pompe, viene attivato un ulteriore relè che separa la batteria dal caricabatterie: questo per evitare che durante il funzionamento di una delle pompe parte della corrente venga prelevata dal caricabatterie.

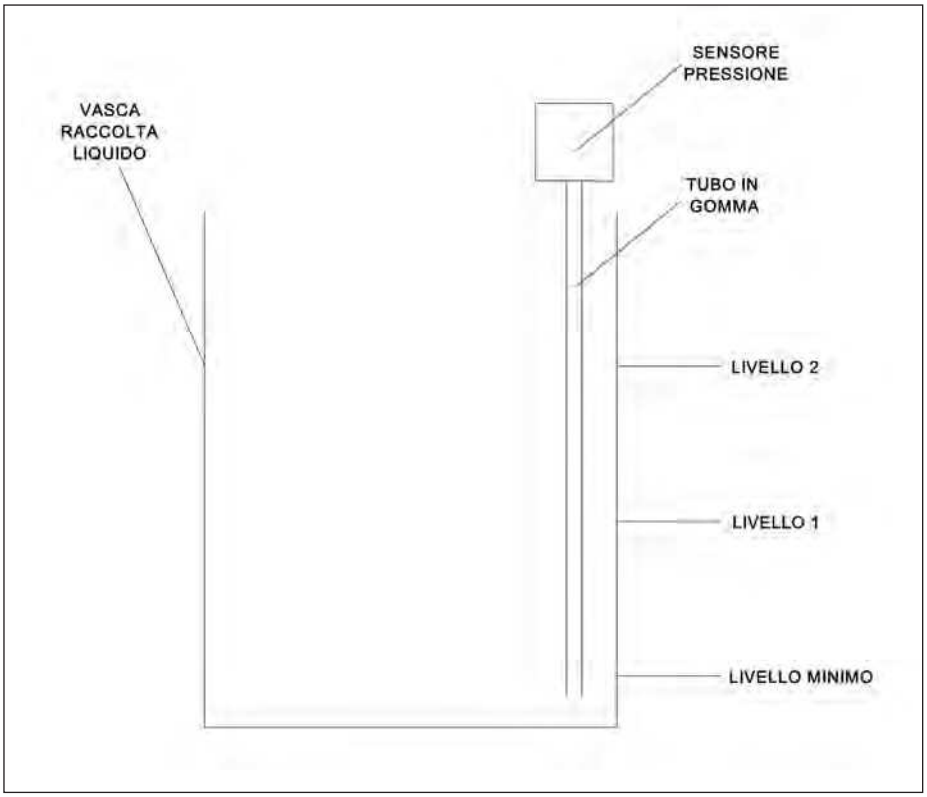
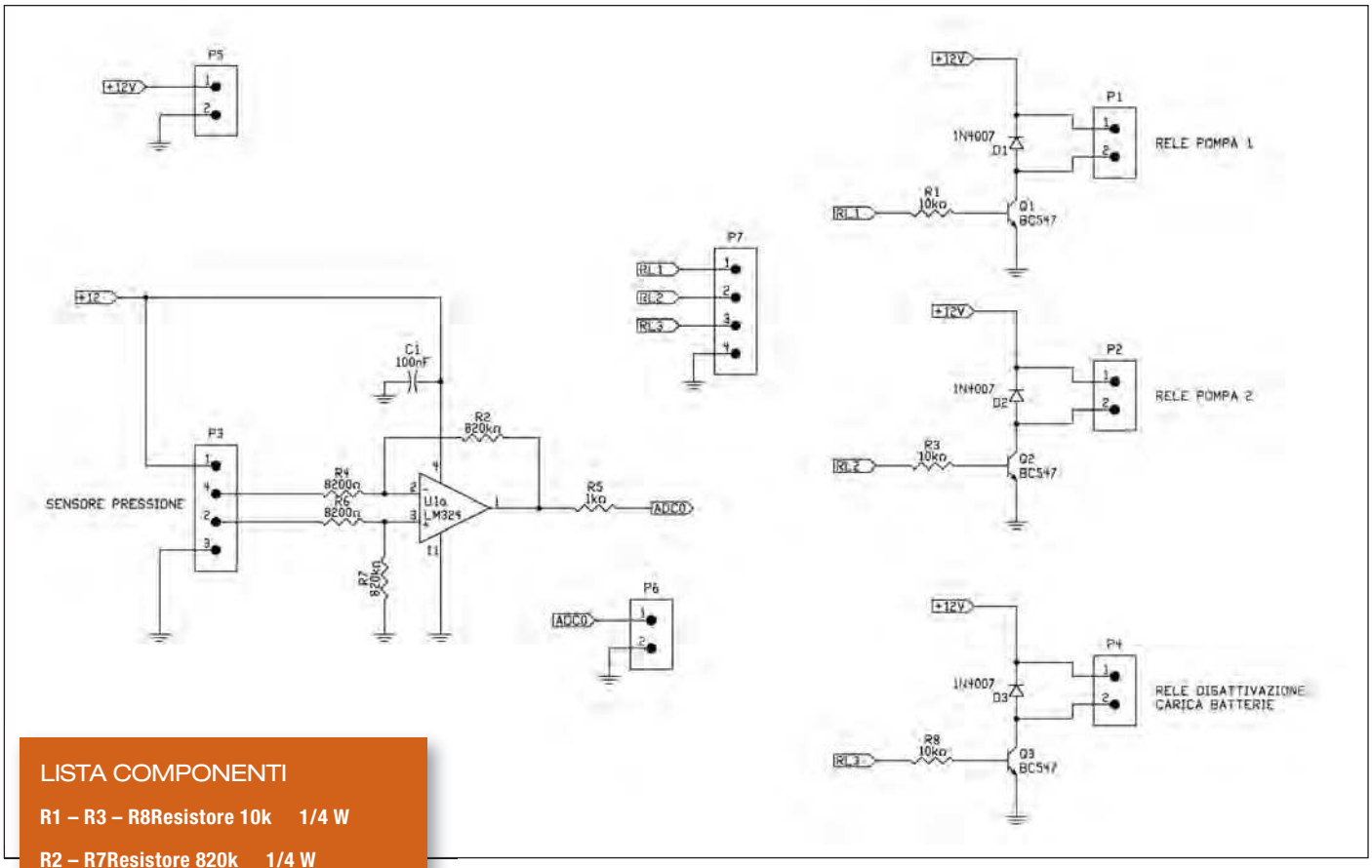


Figura 4: schema montaggio sensore di livello.



LISTA COMPONENTI

R1 – R3 – R8Resistore 10k 1/4 W

R2 – R7Resistore 820k 1/4 W

R4 – R6Resistore 8,2k 1/4 W

C1 Condensatore 100nF

D1 – D2 – D3Diodo 1N4007

Q1 – Q2 – Q3Transistor BC547

U1 Integrato LM324

Figura 5: schema elettrico .

ANALISI DEL CIRCUITO

In **figura 5** è possibile vedere lo schema elettrico del circuito. Il sensore di pressione impiegato dispone di una uscita a ponte di Weatstone: fra i pin 2 e 4 sarà presente una differenza di potenziale proporzionale al valore di pressione misurato. Nel nostro circuito tale differenza di potenziale viene misurata da un amplificatore differenziale con guadagno 100 (820 kohm/8,2 kohm). Come operazionale è stato utilizzato un LM324 (che può funzionare tranquillamente con tensione di alimentazione singola); il condensatore C1 da 100nF serve per prevenire auto oscillazioni dell'operazionale. Il sensore di pressione utilizzato ha un fondo scala di 1 psi che corrisponde ad una

Figura 6: schema elettrico parte di potenza

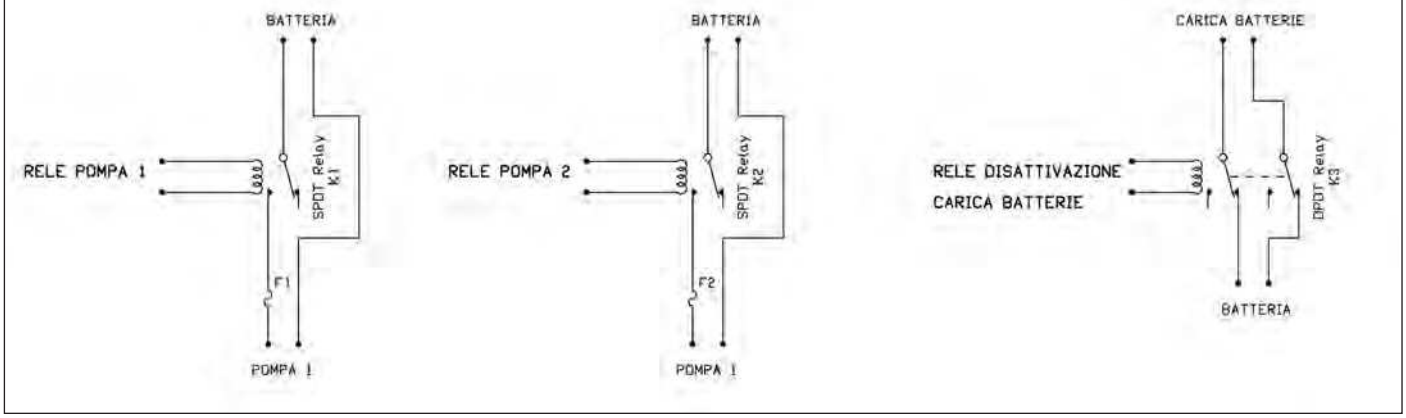
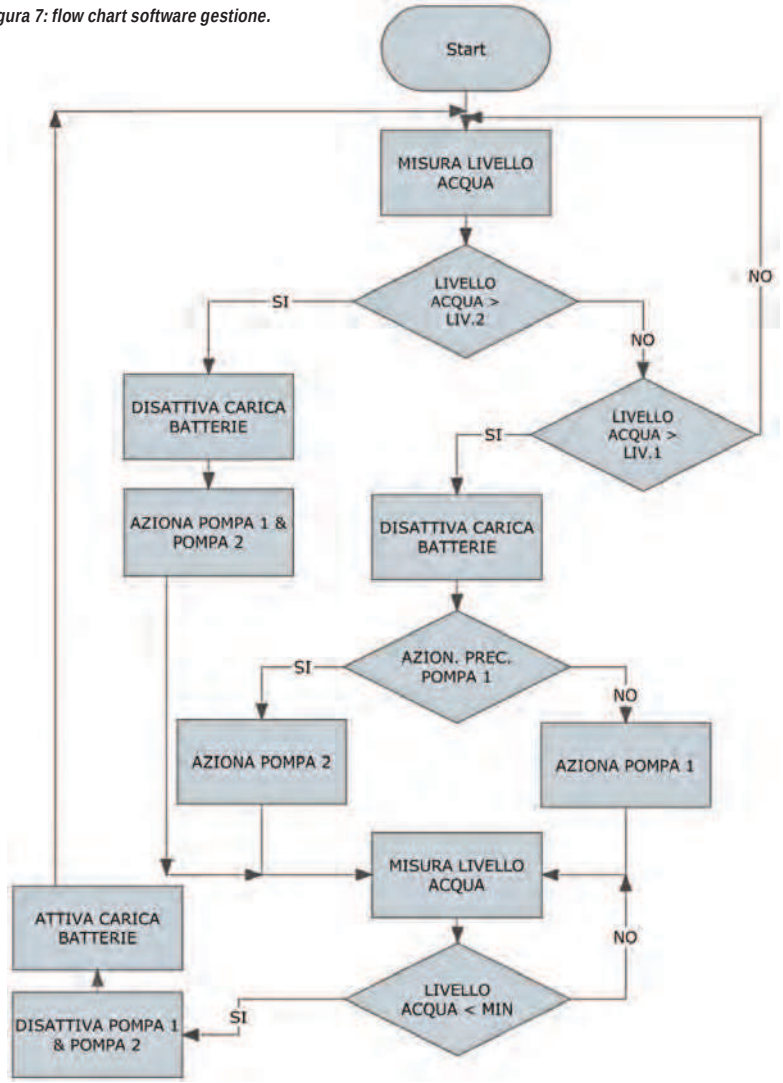
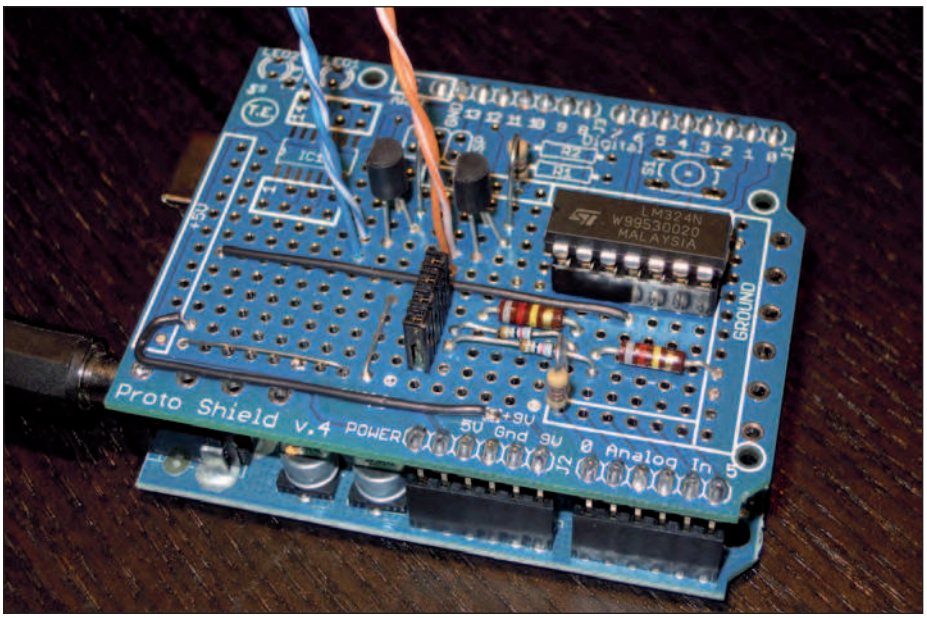


Figura 7: flow chart software gestione.





differenza di potenziale di 45 mV: tale differenza di potenziale viene amplificata di 100 volte in modo da adattarsi all'ingresso analogico della scheda Arduino (0-5V). Quindi, quando il sensore misurerà 1 psi (che corrispondono circa a 700 mm di colonna d'acqua), all'uscita del nostro amplificatore differenziale avremo circa 4,5 V. La tensione in uscita dall'amplificatore differenziale viene applicata ad uno degli ingressi analogici dell'Arduino per essere convertita in digitale. Nel circuito sono poi presenti dei transistor NPN BC547 che servono per eccitare sia i relè di attivazione delle pompe che quelli di disabilitazione del circuito di ricarica delle batterie. E' sufficiente portare a livello logico 1 l'uscita digitale dell'Arduino per mandare in saturazione il transistor e, quindi, eccitare il relativo relè. I diodi 1N4007 posti in parallelo alle bobine dei relè, servono a tagliare le sovratensioni che si creano quando il relè stesso viene disattivato. Tutto il circuito preleva la tensione di alimentazione di 12V dalla scheda Arduino. In **figura 6** è possibile vedere lo schema di collegamento della parte di potenza. In serie al morsetto positivo di alimentazione delle pompe, è consigliabile inserire un fusibile di protezione (il dimensionamento del fusibile andrà fatto in base alla pompa scelta). Facciamo notare che il relè che disattiva il carica batterie è un doppio scambio (in tal modo viene sconnesso sia il positivo che il negativo del carica batterie) e si utilizzano i contatti normalmente chiusi: quindi, quando il relè non è eccitato la batteria risulta essere sotto carica; quando il relè è eccitato, il carica batterie viene disconnesso. Come relè va quindi utilizzato un relè da 16A a doppio scambio.

SOFTWARE GESTIONE

In **figura 7** è riportato il flow chart del software di gestione. Il software misura il livello dell'acqua: se è superiore al livello 2, è necessario attivare contemporaneamente entrambe le pompe (solo dopo aver disattivato il caricabatterie). Se, invece, il livello dell'acqua è superiore al livello 1 viene attivata un'unica pompa.
In questo caso se nell'ultima attivazione

Figura 8: prototipo.

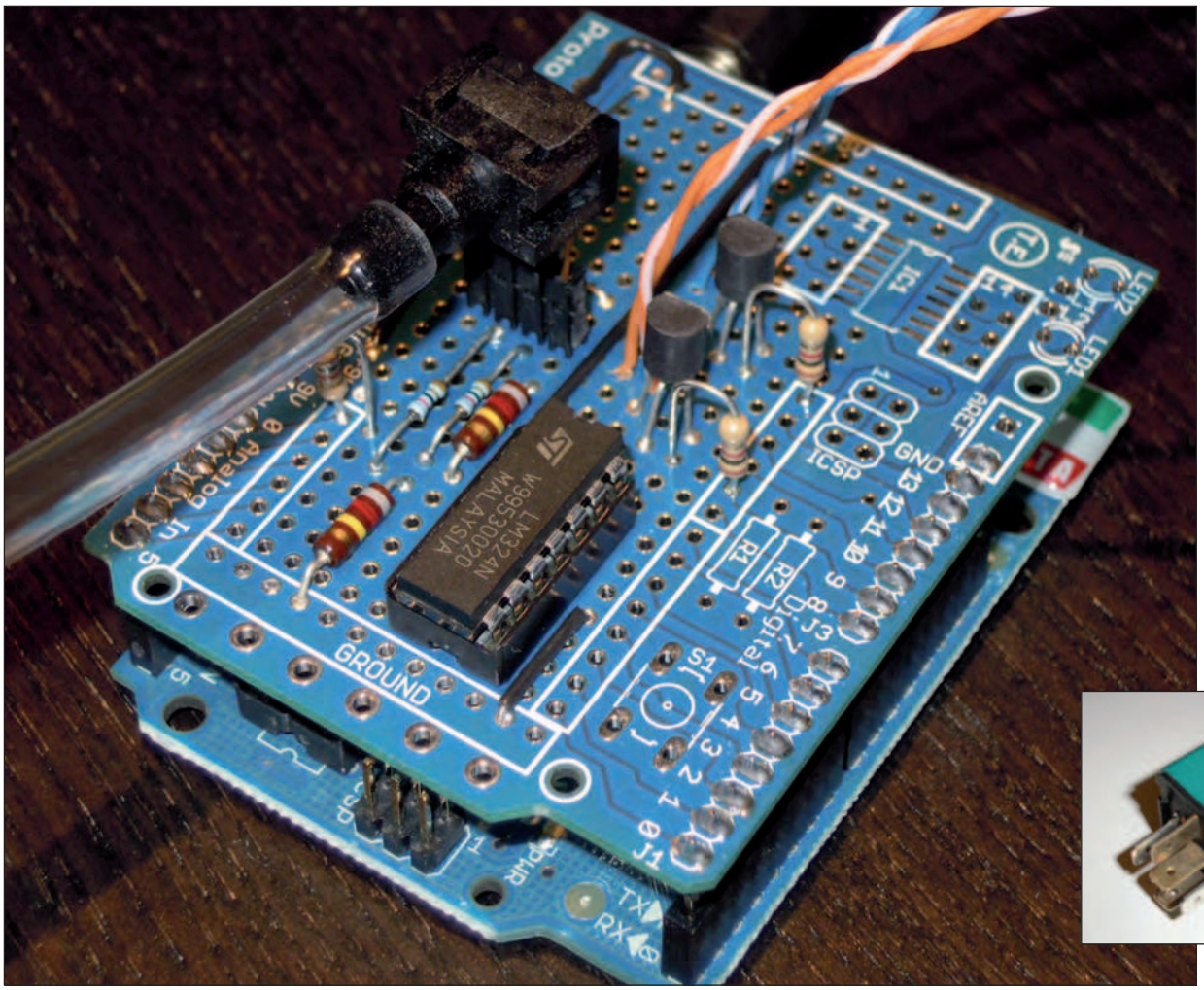


Figura 9: prototipo con sensore montato.

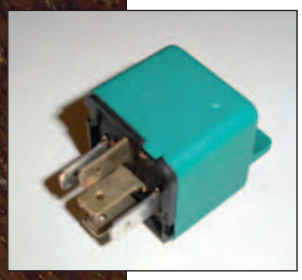


Figura 10: relè.

era stata usata la pompa 1 viene utilizzata la pompa 2 altrimenti viene utilizzata la pompa 1 (questo proprio per far funzionare in maniera alternata le due pompe evitando che una delle due non venga mai utilizzata). Prima di azionare una delle due pompe viene sempre disattivato il caricabatterie. Il software poi misura il livello dell'acqua fino a quando non scende sotto al livello minimo: in questo caso vengono disattivate entrambe le pompe e riattivato il carica batterie. A questo punto il ciclo riparte. Per come è stato impostato il software, il livello minimo corrisponde a circa 10 cm di acqua, il livello 1 a 30 cm e il livello due a 60 cm. Tali valori possano essere facilmente cambiati modificando il valore di alcune variabili all'interno del software.

REALIZZAZIONE PRATICA

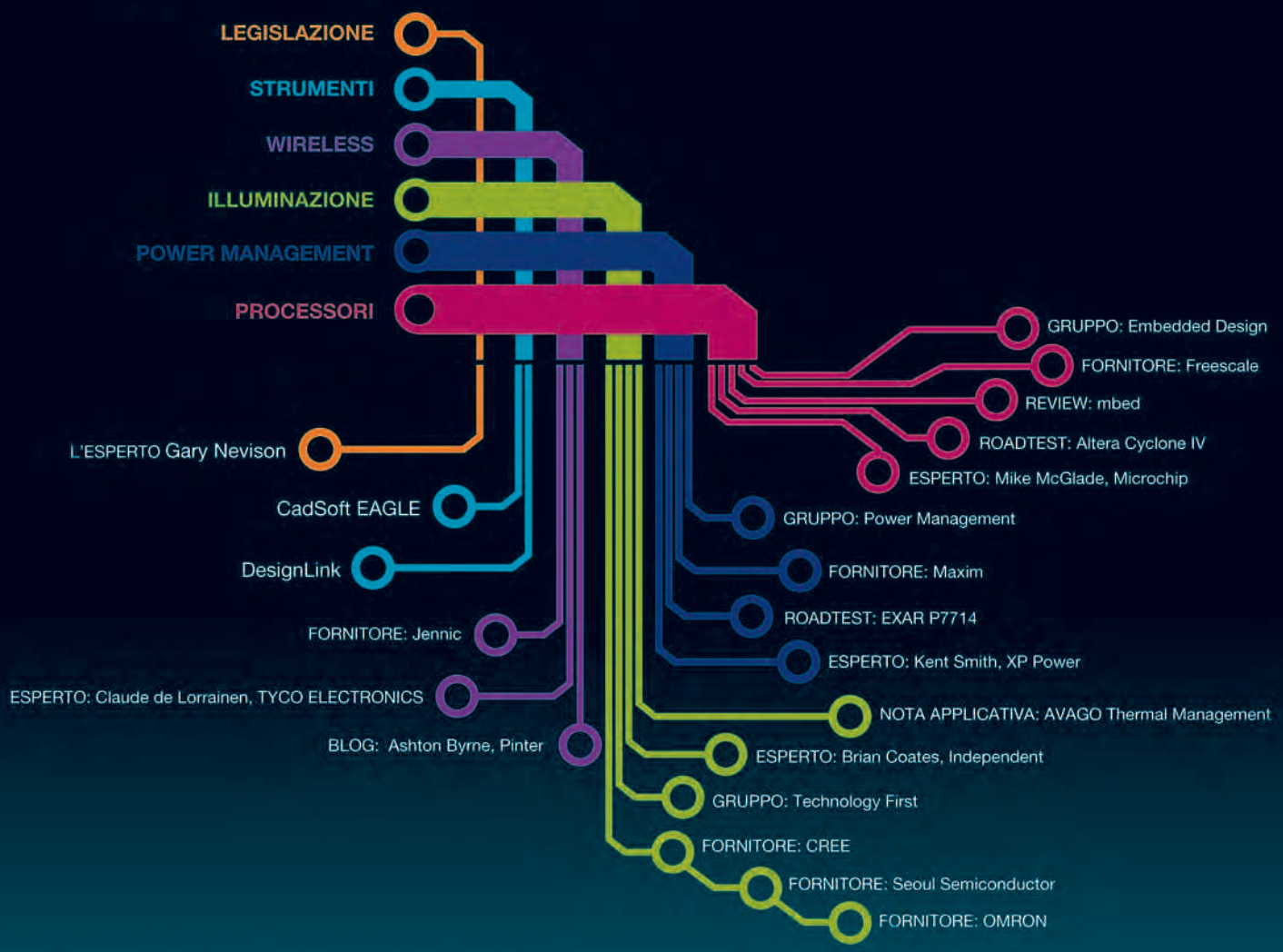
Data la semplicità del circuito per la realizzazione pratica si è deciso di utilizzare lo shield (una scheda millefori adatta al montaggio sopra alla scheda Arduino). In **figura 8** viene riportata una foto del prototipo. In **figura 9** è possibile vedere il prototipo con il sensore montato: è possibile montare il sensore direttamente sopra alla scheda (come in foto) oppure esternamente alla scheda e collegarlo tramite un cavetto a 4 fili.
Per quanto riguarda i relè è stato deciso di utilizzare dei modelli con connessioni a faston: questo tipo di componente presenta il vantaggio di poter essere sostituto molto facilmente ed evita che le elevate correnti che alimentano le pompe passino attraverso la scheda. Inoltre, è facile adattare il relè alla corrente assorbita dalla pompa. Si consiglia di inserire la scheda elettronica e tutta la parte di potenza (relè e fusibili) all'interno di un contenitore stagno.

CONCLUSIONI

Con questo progetto abbiamo visto che con una spesa relativamente ridotta e un po' di impegno si riesce a realizzare un valido sistema per lo svuotamento di pozzetti di raccolta acqua.
Il sistema, tra l'altro, può essere facilmente modificato per essere adattato alle più svariate condizioni di impiego: come al solito l'unico limite è la nostra fantasia! ❑

CODICE MIP 2804981

CONNETTITI



LA TUA LINEA DIRETTA CON I PRINCIPALI BRAND DELL'ELETTRONICA

Entra a far parte della community element14 e troverai una grande quantità di informazioni tecniche fornite direttamente da brand leader a livello mondiale. La community è il posto giusto per metterti in contatto e interagire in modo diretto con esperti progettisti elettronici, che terranno discussioni e risponderanno alle domande dei membri della community. element14 è ad oggi la più ampia community di progettisti di tutto il mondo e il portale più influente dell'industria elettronica.

REGISTRATI SU www.element14.com

www.element14.com

element14
GET YOURSELF CONNECTED

CODICE MIP 2807442

di LUCA STANZANI

Allarme SMS per AUTO

In questo articolo cerchiamo di progettare un sistema di antifurto per automobili in grado di avvertire il proprietario del mezzo del tentativo di furto, ma non solo! Il circuito dovrà essere in grado di segnalare la posizione dell'auto ovunque questa si trovi. Cosa c'è di meglio allora che sfruttare la rete cellulare GSM che copre quasi la totalità del territorio nazionale?

Il problema dei ladri d'auto è uno dei più sentiti da parte degli automobilisti di tutto il mondo. Solamente una parte limitata di antifurti è in grado di avvertire il proprietario che la propria auto è stata rubata e nella maggior parte solamente se questo si trova non troppo distante (500-800m). Esistono dispositivi per tracciare la posizione dell'auto sfruttando i segnali GPS, ma a questo punto si va su sistemi d'allarme estremamente costosi. L'ideale sarebbe avere uno strumento che ci avverta del furto senza limitazioni di range d'azione e che consenta al proprietario dell'auto di operare su alcuni controlli dell'auto dopo che questa è stata rubata.

LA SOLUZIONE DEL PROBLEMA

Per rispondere ai problemi descritti in precedenza, una soluzione applicabile è quella di sfruttare gli SMS. Il servizio di messaggistica breve è offerto da praticamente quasi tutti gli operatori telefonici e sono sempre di più le aziende che sviluppano sistemi di sicurezza SMSSS (SMS-based Security System) che per-

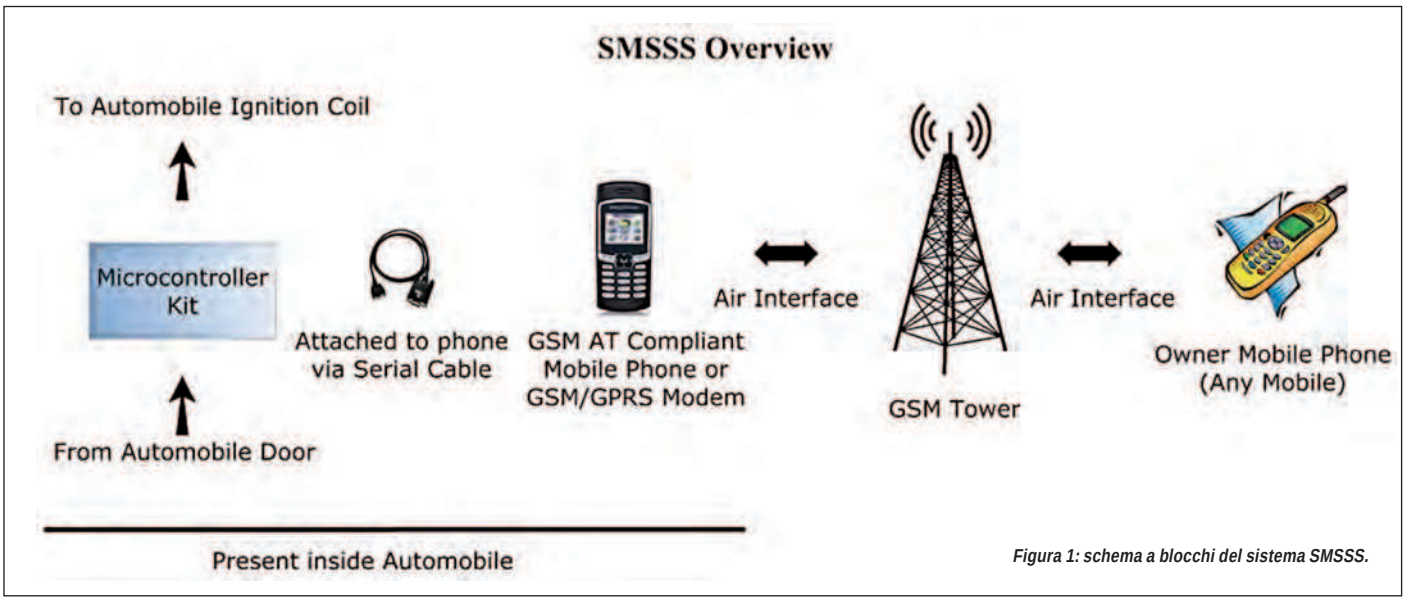


Figura 1: schema a blocchi del sistema SMSSS.

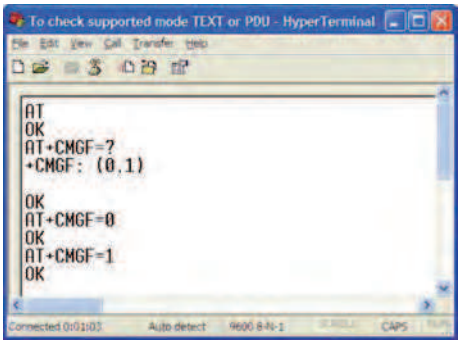


Figura 2: controllo delle modalità di invio SMS supportate dal telefono

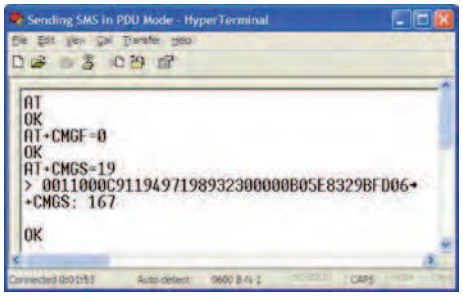


Figura 3: invio SMS in modalità PDU.

TABELLA 1

AT COMMAND	SIGNIFICATO
+CMGF	Definisce il formato del messaggio (Testo o PDU)
+CPMS	Seleziona l'area di memorizzazione
+CMGW	Scrive il messaggio in memoria
+CMSS	Manda un messaggio salvato in memoria
+CMGR	Legge un messaggio
+CMGD	Cancella un messaggio
+CGREG	Registrazione alla rete

Tabella 1: comandi AT per il sistema SMSSS.

mettono di avere il controllo sull'automezzo in un'area vastissima. Il sistema SMSSS va installato sull'auto da controllare, per esempio potrebbe monitorare l'apertura degli sportelli. Allo "scatenarsi dell'evento", un SMS viene inviato sul telefono cellulare del proprietario dell'auto, avvisandolo del possibile furto. Dopo la ricezione del messaggio, il proprietario dell'auto può rispondere con un ulteriore messaggio che contenga una password per immobilizzare il veicolo. Dopo la ricezione dell'SMS, il sistema SMSSS potrebbe agire sull'iniezione dell'automezzo impedendo che possa essere messo in moto. Allo stesso tempo, potrebbe continuare ad informare il proprietario con l'esatta ubicazione dell'automezzo fornita dall'antenna telefonica a cui è collegato. Oppure questo meccanismo potrebbe essere invocato dal proprietario dell'auto sempre con un SMS. Quanto descritto è schematizzato in **figura 1**. Il telefono GSM, contenuto all'interno dell'auto, si può interfacciare direttamente con un microcontrollore AVR attraverso la porta seriale. Il telefono GSM può essere controllato inviandogli dei comandi AT dal microcontrollore.

MODEM GSM E INTERFACCIA

Il sistema SMSSS utilizza un modem GSM per inviare e ricevere gli SMS dal proprietario del mezzo. Il dialogo avviene attraverso la circuiteria a bordo del microcontrollore, ma un requisito indispensabile è che il telefono cellulare collegato al sistema SMSSS sia equipaggiato con un modem GSM/GPRS. La velocità di trasmissione delle informazioni non cambia, poiché questo parametro è funzione

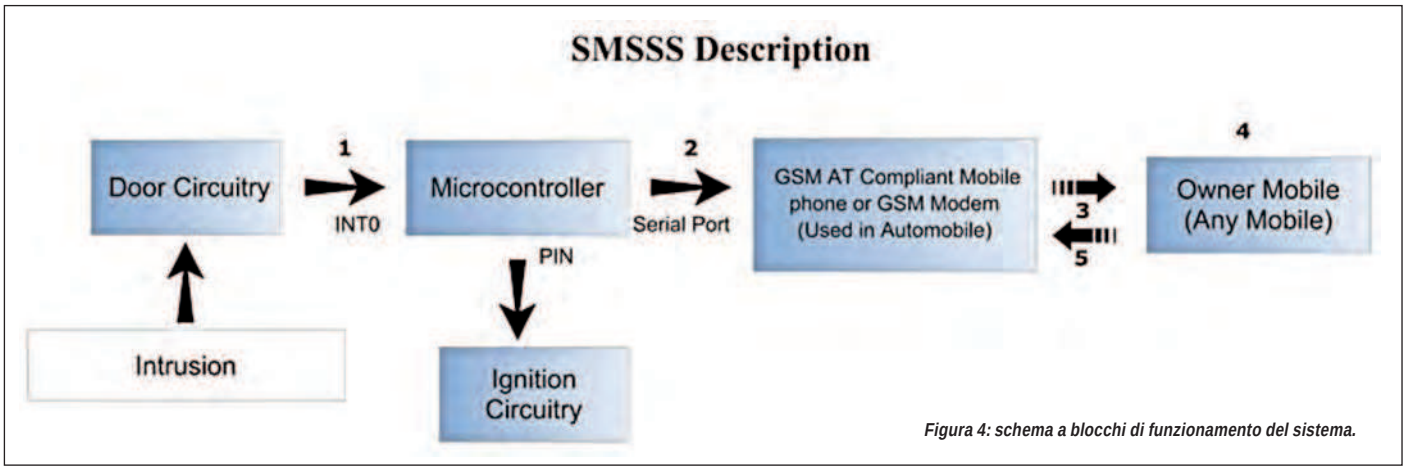


Figura 4: schema a blocchi di funzionamento del sistema.

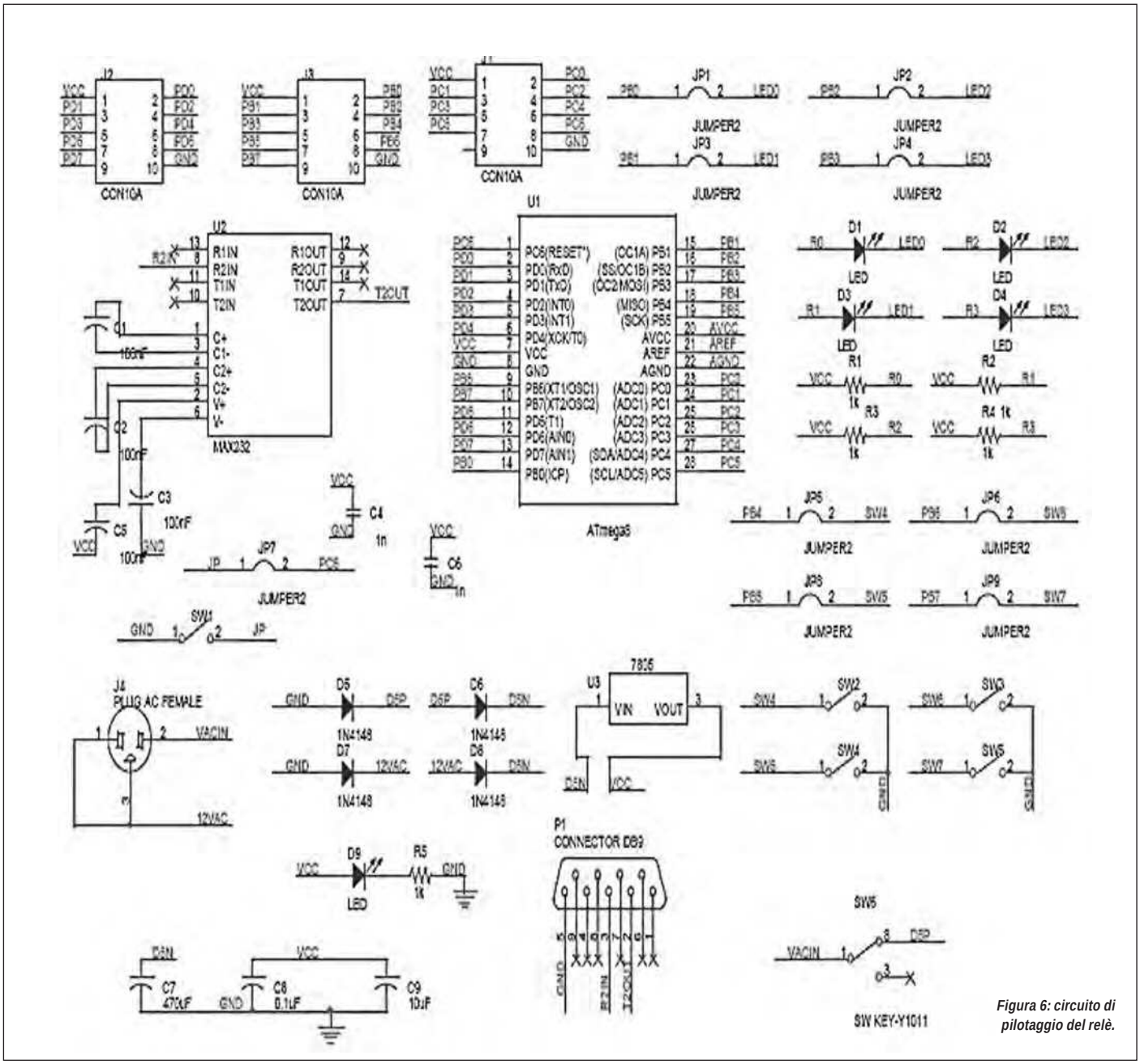


Figura 6: circuito di pilotaggio del relè.

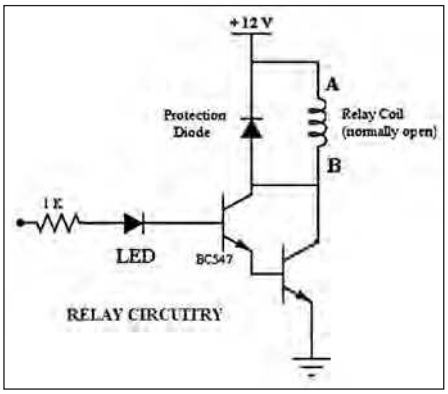


Figura 5: schema elettrico del sistema SMSSS.

della rete wireless. La presenza del modem GSM consente al microcontrollore di poter inviare al telefono dei comandi detti AT (Attention). In **tabella 1** sono riportati una serie di comandi AT utilizzati dal sistema. Ci sono due modi per inviare un messaggio SMS, attraverso la modalità testo oppure in modalità PDU (Protocol Description Unit), la differenza risiede nel set di caratteri codificati che nel caso testo è più limitato. Per questo, il sistema SMSSS utilizza la modalità PDU per non avere limiti di codifica. In **figura 2** è mostrato come vedere quale modalità è supportata dal telefono o dal modem GSM/GPRS. In **figura 3** e sono mostrati degli esempi di invio e risposta a comandi AT utilizzando il programma HyperTerminal per l'invio di messaggi SMS in PDU mode.

MICROCONTROLLORE E CIRCUITI DI INTERFACCIA

Il microcontrollore utilizzato nel nostro sistema SMSSS è un ATmega8. Riceve in ingresso sul pin 0 della porta C, un segnale dal circuito di apertura delle porte. L'arrivo del segnale genera un interrupt che invia un messaggio SMS di allerta al proprietario dell'automobile, utilizzando il modem GSM collegato ai pin 0 (RX) e al pin 1 (TX) della porta D. Viene anche pilotato il pin 0 della porta B collegato ad un relè per un circuito di immobilizzazione del mezzo dopo la ricezione di un SMS dal proprietario. Per realizzare quanto descritto, è necessario inizializzare la UART dell'ATmega8 come di seguito:
USART Control and Status Register B: impostato inizialmente a 00011000 per abilitare trasmissione e ricezione.

plug in
I NOSTRI CIRCUITI STAMPATI HANNO DEI COMPONENTI UNICI. SU MISURA PER VOI.
SINGOLA E DOPPIA FACCIA CON FORO METALLIZZATO; CON SOLDER E SERIGRAFIE PER UNO STAMPATO DI ALTA QUALITÀ O SOLO PISTE STAGNATE PER UN PROTOTIPO A BASSO COSTO
velocità
TEMPO DI CONSEGNA GARANTITO: 24 ORE O I CIRCUITI SONO GRATIS
qualità
I CIRCUITI VENGONO SOTTOPOSTI A TEST ELETTRICO E ACCOMPAGNATI DAL CERTIFICATO DI GARANZIA*; TUTTI I CIRCUITI SONO REALIZZATI SU LAMINATO Panasonic
novità
DA OGGI, ALLO STESSO PREZZO E ALLE STESSE CONDIZIONI DEI NOSTRI CIRCUITI DOPPIA FACCIA È DISPONIBILE IL LAMINATO IN ALLUMINIO AD ALTISSIMA DISSIPAZIONE TERMICA INDISPENSABILE PER LA REALIZZAZIONE DI PROGETTI CON LED AD ALTA LUMINOSITÀ E OVUNQUE SIA RICHIESTO UN ELEVATO GRADO DI DISSIPAZIONE
CODICE MIP 2794796
facilità
PER CHIARIMENTI, DETTAGLI SULLE NOTE TECNICHE, ORDINI:
www.mdsrl.it
PREVENTIVO ANONIMO, GRATUITO E IMMEDIATO
*la certificazione sarà rilasciata solo per circuiti con consegna oltre le 48 ore
md millennium dataware
produciamo circuiti stampati
Parco scientifico e tecnologico
15050 Rivalta Scrivia - Tortona (AL)
tel. 0131 860.254 fax 0131 860157 info@mdsrl.it

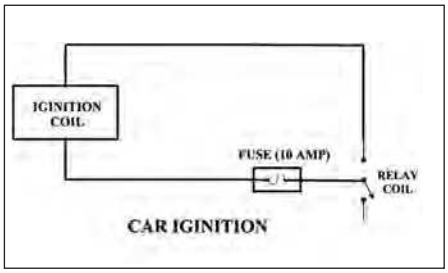
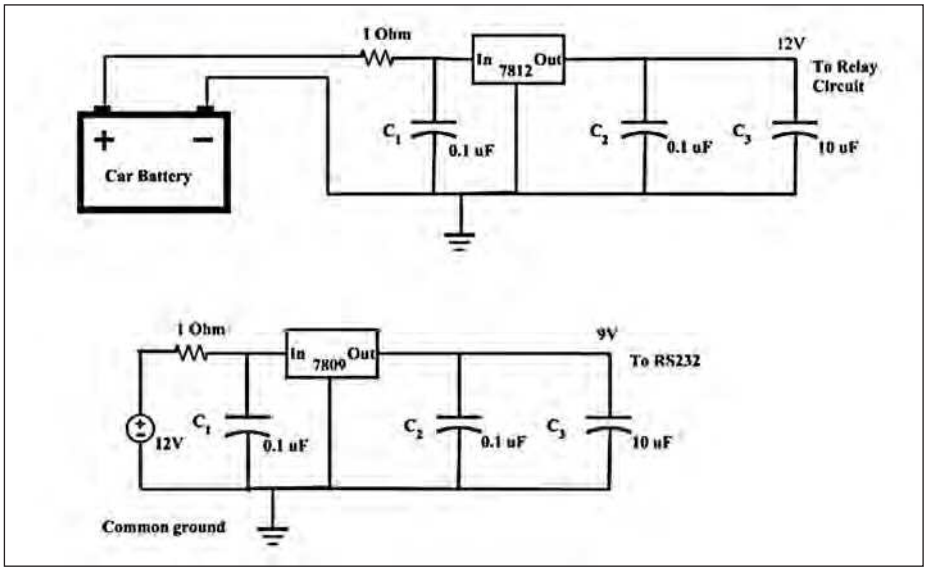


Figura 7: schema per interrompere l'alimentazione degli iniettori.

Figura 8: circuiti elettrici per la generazione delle tensioni di alimentazione.



• USART Control and Status Register C: impostato inizialmente a 10000110 per definire che la USART è in modalità asincrona, senza bit di parità, uno stop bit e 8 data bit. USART BAUD RATE REGISTERS

• UBRRL impostato a 51 e UBRRH impostato a 0. La baud rate cosi ottenuta è 9600bps. Oltre a queste operazioni bisogna procedere con:

MCU Control Register: da impostare a 00000000. interrupt generati sul livello basso del segnale collegato al pin INT0.

General Interrupt Control Register – GICR: da impostare a 01000000 per abilitare gli interrupt su INT0.

Stack pointers (SPL & SPH) sono inizializzati alla RAM END.

DESCRIZIONE DEL SISTEMA

In **figura 4** è possibile osservare lo schema a blocchi di funzionamento del sistema. Appena la porta dell'automobile viene aperta, il pin PD2 (INT0) del microcontrollore passa dal livello alto al livello basso. La routine di interrupt parte e invia un dato prememorizzato utilizzando il comando "AT+CMSS=1" diretto verso il telefono GSM collegato alla porta seriale del microcontrollore. Alla ricezione dei dati sulla porta seriale, il telefono accede al messaggio memorizzato nella locazione 1 e lo invia al numero salvato nelle preferenze del messaggio. Il numero a cui inviare il messaggio sarà quello del proprietario dell'auto, che riceverà la segnalazione e la richiesta della password per il blocco dell'iniezione dell'automobile. In risposta il proprietario dell'auto invia un codice numerico di 4 cifre al telefono/Modem GSM presente nel sistema SMSSS sull'auto. Questo codice numerico di quatto cifre può essere una "Immobilize password" per bloccare l'avvio della macchina ma può anche essere una "Location password", in tal caso si riceve l'ubicazione del veicolo tramite l'informazione restituita dalla rete telefonica cellulare. Tale informazione si ottiene con il comando "AT+CGREG?", la risposta è memorizzata nella locazione 2 di memoria del telefono e viene inviata al numero che ha richiesto inviato la "Location password". L'SMS ricevuto viene per default memorizzato nella prima locazione libera del telefono inglobato nel sistema SMSSS (nel nostro caso la 3). Il microcontrollore perciò controlla periodicamente la locazione 3 con il comando "AT+CMGR=3" se si ha risposta positiva, allora il contenuto viene letto e la password ricevuta confrontata con quelle memorizzate nel codice. Dopodiché ogni messaggio in entrata viene eliminato. E il ciclo riprende.

CONCLUSIONI

Il sistema di allarme SMSSS progettato è estremamente efficace nelle situazioni di furto o tentativo di furto dell'automobile e costituisce una base di sviluppo dalla quale partire per perfezionare o personalizzare il sistema. Per esempio, oltre al segnale di apertura delle portiere, si possono controllare anche le vibrazioni indotte sull'automezzo, inserendo un sensore di vibrazione che cambi stato logico e che produca lo stesso risultato di invio di un SMS di avvertimento al proprietario. La stessa filosofia di tagliare l'alimentazione agli iniettori in seguito alla ricezione di un SMS può essere estesa ad altre operazioni, lasciando piena libertà ai progettisti. Vale la pena considerare che progettare un sistema d'antifurto di questo tipo è sicuramente più economico rispetto all'acquisto di uno equivalente presente sul mercato, in più si ha la possibilità di personalizzare il sistema a piacimento. In **figura 5** c'è lo schema completo del sistema, mentre nelle **figure 6**, **7**, **8** è possibile osservare rispettivamente i circuiti per il relè, per gestire l'interruzione dell'alimentazione agli iniettori e per generare le alimentazioni necessarie al funzionamento. ❑

CODICE MIP 2804871

PER *approfondire...*

- Ankur Verma, "SMS for automobile security system developers".
- AVR Atmega8 Datasheet, disponibile da www.atmel.com
- Comandi AT, http://it.wikipedia.org/wiki/Comandi_AT

entra in un mondo di contenuti digitali

CLUB DELLE RIVISTE

E-BOOK

ANNATE IN PDF

fare elettronica DIGITAL EXTRA

CODICE MIP 2808429

il riferimento per tutta l'elettronica sul tuo PC o TABLET

E-BOOK

I migliori libri di elettronica in formato e-book.
Puoi leggere e stampare i capitoli o le pagine di interesse (ad alta risoluzione) e installare i software eventualmente trattati nei volumi.

ANNATE IN PDF

Le annate complete delle riviste Fare Elettronica, Firmware ed Elektor in comodi file zip navigabili o masterizzabili su CD o DVD completi di tutte le riviste (e relativi downloads) uscite nei vari anni, da non perdere!

CLUB DELLE RIVISTE

Scarica illimitatamente per un anno centinaia di articoli pubblicati sulle riviste Fare Elettronica e Firmware.

fare elettronica DIGITAL EXTRA

Monografie di elettronica in formato digitale: raccolte di progetti completi, CNC e microcontrollori.

www.ieshop.it

INWARE EDIZIONI

di GIUSEPPE LA ROSA

DISSUASORE AUDIO con messaggi vocali

Dissuasore audio abbinabile a qualsiasi sensore a doppia tecnologia. Attivabile e disattivabile tramite centrale di allarme, con uscita di attivazione allarme e di relè per lampeggiante, possibilità d'uso anche (da solo) senza centrale d'allarme

Il sistema che vi proponiamo, è stato progettato per riprodurre un messaggio pre-registrato, quando un qualsiasi sensore a doppia tecnologia rileva movimento di persone o veicoli. Questo progetto permette una vasta gamma di applicazioni, come messaggi di benvenuto agli ingressi dei negozi, messaggi di allarme o di avvertimento nei cantieri e in zone pericolose, avvisi di superamento aree riservate. Ma prima di passare alla descrizione dettagliata del progetto, facciamo una breve descrizione del Voice-Recorder ISD2560: un riproduttore e registratore allo stato solido usato in questo progetto.

IL REGISTRATORE E RIPRODUTTORE ISD2560

Per realizzare il dissuasore audio, bisogna realizzare un circuito che sia capace di registrare, memorizzare e riprodurre (su comando di sensori a doppia tecnologia) in uno spazio molto limitato, senza per questo sacrificare la qualità del segnale audio registrato.

Figura 1: foto del progetto finito.

Occorreva, dunque, un integrato che ci consentisse di svolgere tutte queste funzioni in modo impeccabile. Queste caratteristiche le abbiamo trovate nell'integrato ISD2560 della Winbond.
Vi riassumiamo qui le sue caratteristiche tecniche:

- Sigla integrato: Voice-Recorder ISD2560;
- Volt alimentazione: 4,5-6Volt;
- Assorbimento a riposo: 1µA;
- Assorbimento in registrazione: 22mA;
- Assorbimento in ascolto: 40mA;
- Massima potenza d'uscita: 50mW;
- Carico consigliato in uscita: 8-16 ;
- Massimo segnale d'ingresso: 50mV p/p;
- Distorsione armonica: 1% su1KHz;
- Minima frequenza registrabile: 30Hz;
- Massima frequenza registrabile: 6KHz;
- Tempo massimo di registrazione: 1 Minuto;

Poiché la massima frequenza registrabile dei più comuni Voice-Recorder si aggira intorno ai 2KHz, chi ascolta questo ISD2560 rimarrà stupito dalla fedeltà di riproduzione del suono e dalla timbrica vocale, perché raggiunge i 6KHz.
Guardando lo schema a blocchi di questo integrato ISD2560 (vedi **figura 3**), scopriamo che dispone di 28 piedini, ognuno dei quali svolge una sua specifica funzione:
Piedini 1-2-3: questi tre piedini devono essere collegati alla massa di alimentazione, come potete vedere nella **figura 2** che riproduce lo schema elettrico;
Piedino 4: questo piedino serve per ottenere la funzione di ascolto ripetitivo o di ascolto singolo. Quando il piedino viene collegato a massa, si ha un ascolto di frase singola. Quando risulta collegato alla tensione positiva di 5V, si ha un ascolto ripetitivo;

Piedini 5-6: questi due piedini devono essere collegati alla massa di alimentazione;
Piedino 8: questo piedino deve essere collegato alla massa di alimentazione;
Piedini 9-10: questi due piedini devono essere collegati alla tensione positiva dei 5V;
Piedino 11: questo piedino non viene utilizzato;
Piedini 12-13: questi due piedini devono essere collegati alla massa di alimentazione;
Piedini 14-15: da questi due piedini esce il segnale BF da applicare all'altoparlante. Dal piedino 14 possiamo prelevare il segnale da applicare sull'ingresso di un qualsiasi stadio finale di potenza;
Piedino 16: questo piedino deve essere collegato alla tensione positiva dei 5V;
Piedino 17: su questo piedino viene applicato il segnale BF che vogliamo registrare;

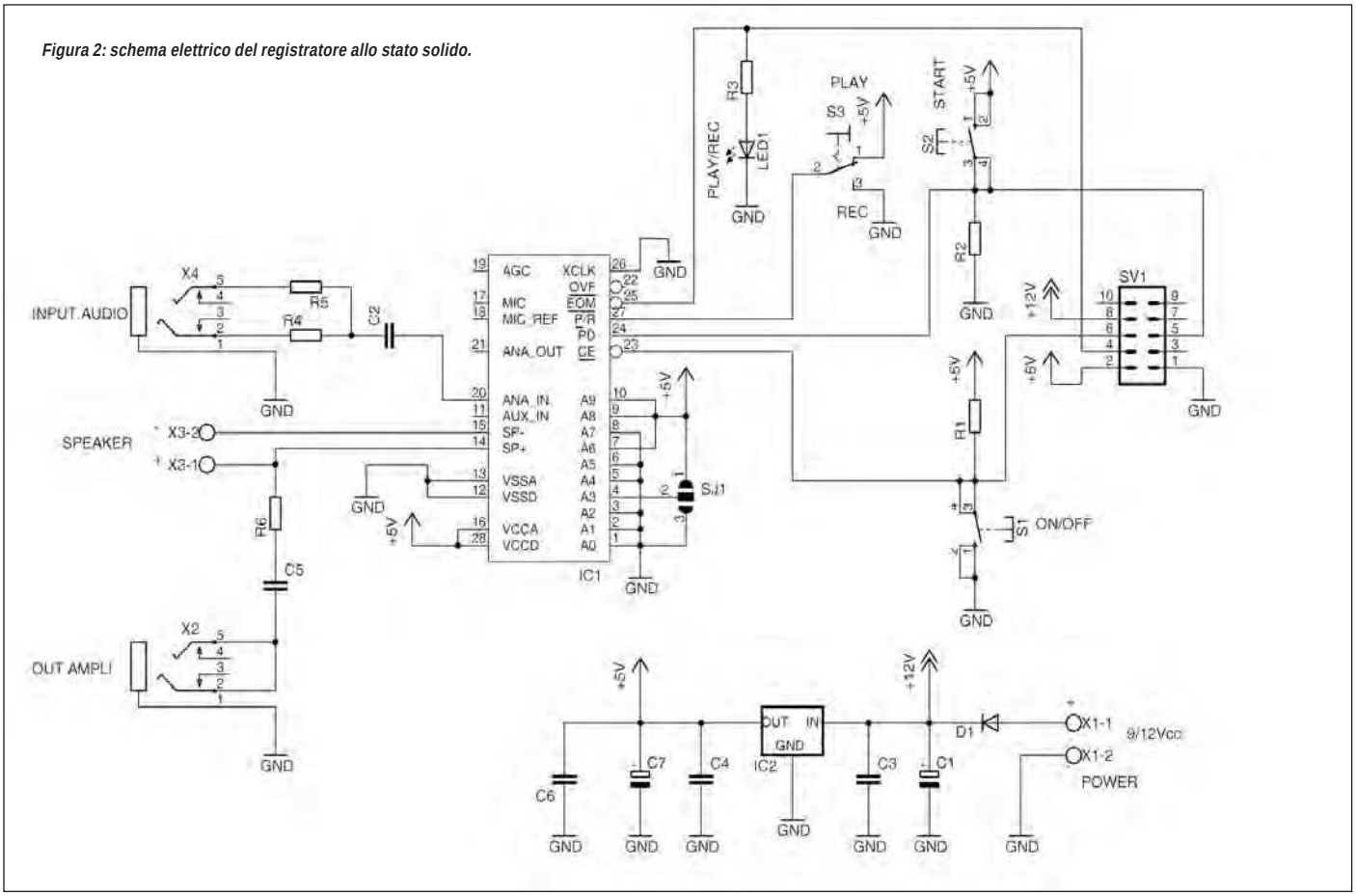


Figura 2: schema elettrico del registratore allo stato solido.

LISTA COMPONENTI

R1-R2	100KΩ 1/4 W
R3	1KΩ 1/4 W
R4-R5	10KΩ 1/4 W
R6	1KΩ 1/4 W
C1	100µF 25V elettrolitico
C2-C3-C4	100nF poliestere
C5	470nF poliestere
C6	100nF poliestere
C7	10µF 25V elettrolitico
D1	1N4007 diodo
IC1	ISD2560
IC2	L7805
LED1	Led rosso 3mm
S1	Pulsante da C.S. nero
S2	Pulsante da C.S. rosso
S3	Deviatore da C.S.
X1	Morsetti a 2 poli
X2	Presa jack stereo 3mm da C.S.
X3	Morsetti a 2 poli
X4	Presa jack stereo 3mm da C.S.
SV1	IDC 10 poli da C.S.
N.1	Zoccolo 14+14pin
N.2	Distanziale C.S. da 3mmm
N.2	Dadi 3mm
N.4	Viti 6x3mm
N.2	Viti auto filettanti 3,5x6mm
N.1	Altoparlante 8Ω 1W diametro 60mm

Tabella 1: elenco componenti registratore allo stato solido.

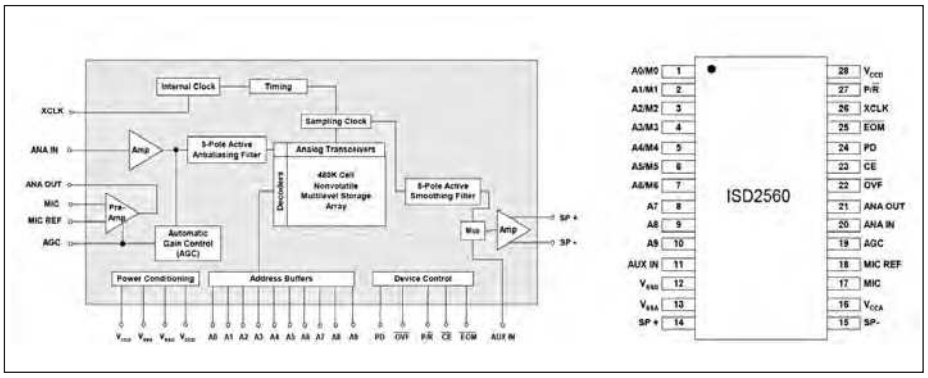


Figura 3: schema a blocchi del Voice-Recorder ISD2560.

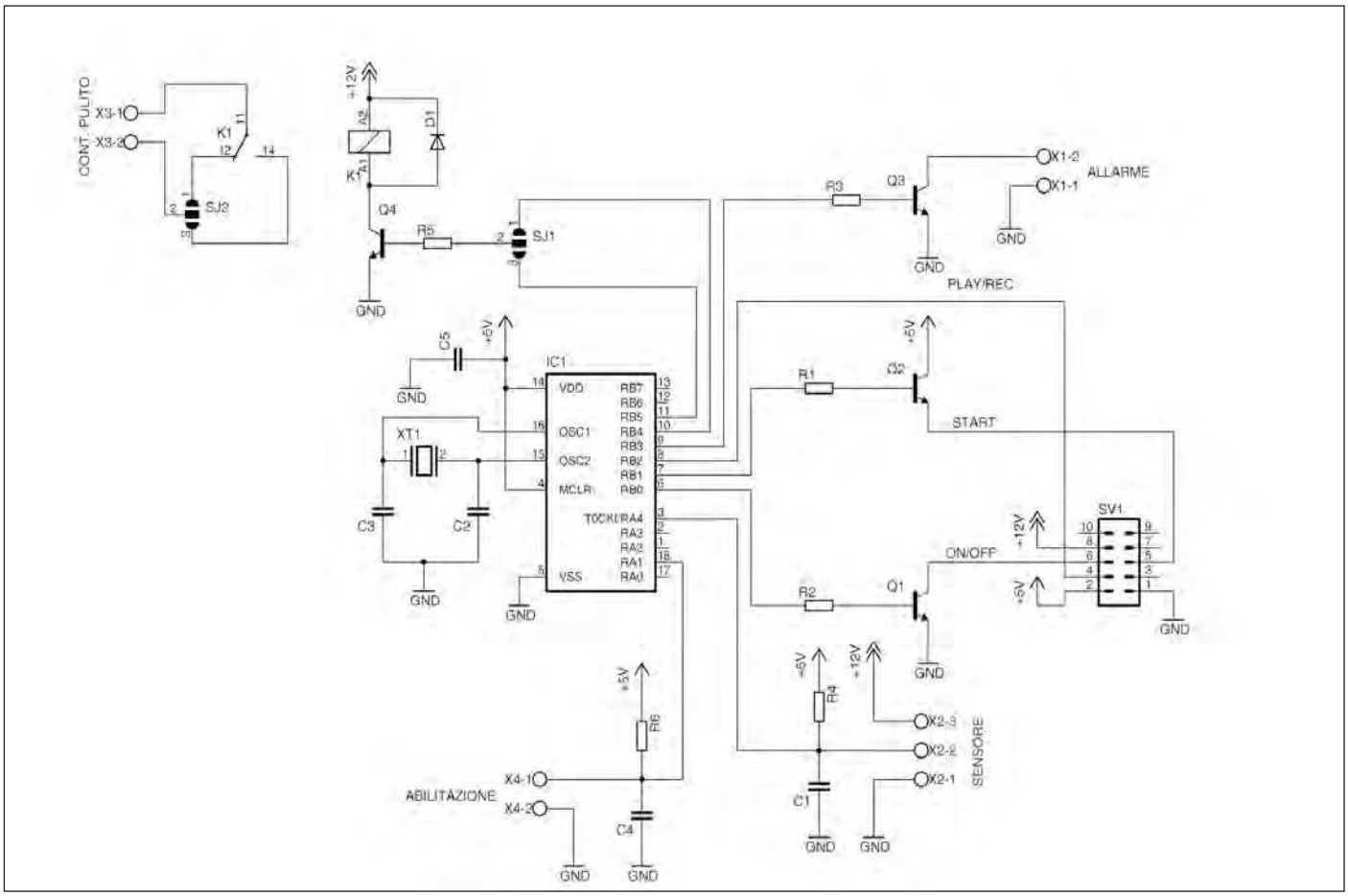


Figura 4: schema elettrico della scheda di controllo.

LISTA COMPONENTI

R1-R2-R3	10KΩ 1/4 W
R4	4,7KΩ 1/4 W
R5	10KΩ 1/4 W
R6	4,7KΩ 1/4 W
C1	10nF poliestere
C2-C3	22pF ceramico
C4	10nF poliestere
C5	100nF poliestere
XT1	4MHZ quarzo
D1	1N4007 diodo
Q1-Q2-Q3-Q4	BC547 transistor
IC1	PIC16F628A
K1	Relè 1 scambio
X1	Morsetti a 2 poli
X2	Morsetti a 3 poli
X3	Morsetti a 2 poli
X4	Morsetti a 2 poli
SV1	IDC 10 poli da C.S.
N.1	Zoccolo 9+9 pin
N.2	IDC 10 poli a crimpare
N.1	Cavo piatto 10cm

Tabella 2: elenco componenti scheda di controllo.

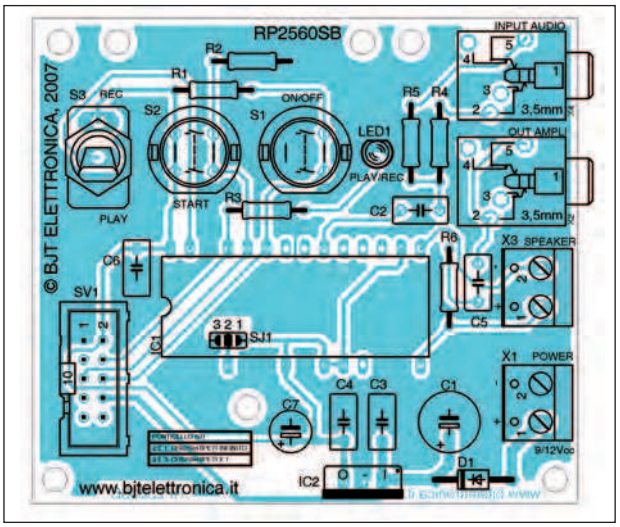


Figura 5: piano di montaggio del registratore allo stato solido.

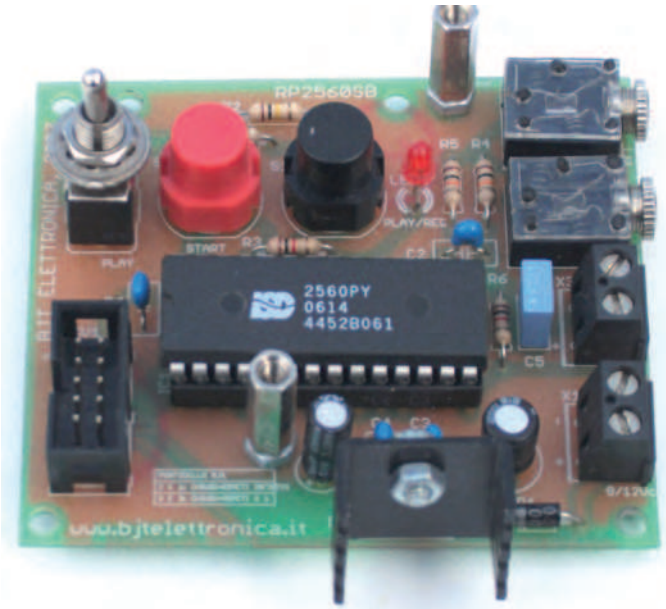
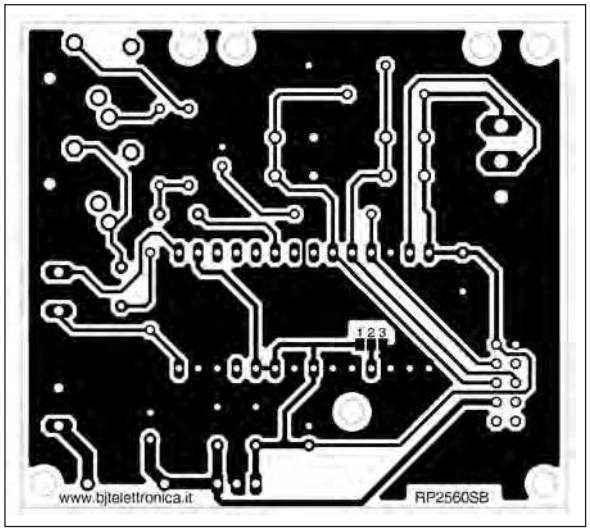


Figura 6: scheda registratore allo stato solido assemblata.

Piedino 18: questo piedino non viene utilizzato;
Piedino 19: questo piedino regola il tempo di intervento dell'AGC;
Piedini 20-21: questi piedini servono per trasferire il segnale prelevato dallo stadio preamplificatore al secondo stadio amplificatore;
Piedino 22: questo piedino non viene utilizzato;
Piedino 23: questo piedino viene utilizzato per la funzione ON/OFF sia durante la registrazione che durante l'ascolto. Per metterlo in pausa, dovrete portarlo a massa una seconda volta;
Piedino 24: questo piedino viene utilizzato come Start in fase di ascolto. Portando questo pin al positivo d'alimentazione, il contatore si porterà all'inizio del sonoro memorizzato.
Piedino 25: da questo piedino esce una tensione positiva di 5V ogni volta che l'integrato viene posto in registrazione o in ascolto. Questa tensione viene utilizzata per accendere un diodo LED;
Piedino 26: questo piedino deve essere collegato alla massa di alimentazione;
Piedino 27: questo piedino viene utilizzato per commutare l'integrato dall'ascolto alla registrazione o viceversa. Commutando questo piedino a massa quando desiderate iniziare a registrare o al positivo d'alimentazione quando si vuole ascoltare.
Piedino 28: questo piedino va collegato alla tensione positiva di alimentazione.

Figura 7: circuito stampato registratore allo stato solido in scala 1:1 (lato rame).



Dopo questa breve descrizione passiamo alla descrizione degli schemi elettrici.

SCHEMA ELETTRICO

Questo progetto si compone di due schemi elettrici: il primo, è quello riguardante il registratore allo stato solido che è un vero registratore e riproduttore audio..
Il secondo schema elettrico, è quello riguardante la scheda di controllo che ha la funzione d'interfacciare il sensore a doppia tecnologia alla scheda del registratore allo stato solido.
La **figura 2** illustra lo schema elettrico del registratore allo stato solido. La presa jack d'ingresso X4 accetta qualsiasi uscita audio proveniente da lettori MP3 o schede audio di PC, da essa è possibile mandare il messaggio audio da registrare al piedino 20 di IC1, le resistenze R4 e R5 miscelano il segnale stereo proveniente da lettori MP3 o altre fonti audio e lo trasformano in un segnale mono che viene applicato al condensatore C2 che, a sua volta, lo disaccoppia e lo applica al piedino 20 di IC1.
Il deviatore S3, il cui piedino centrale è collegato al piedino 27 di IC1, se viene chiuso verso massa imposta IC1 come registratore (Rec).
Se viene chiuso verso il positivo di alimentazione, cioè +5V, imposta IC1 come riproduttore (Play).
Se S3 è chiuso verso massa (modalità registrazione) e il pulsante S1 viene premuto, il piedino 23 di IC1 viene chiuso a massa, il che consente di eseguire la registrazione. Premendo nuovamente S1, durante la registrazione, questa viene messa in pausa. Se il deviatore S3 è chiuso al positivo di alimentazione (modalità di riproduzione) e viene premuto S1 si avvia la riproduzione dell'audio. La riproduzione viene messa in pausa premendo nuovamente S1.

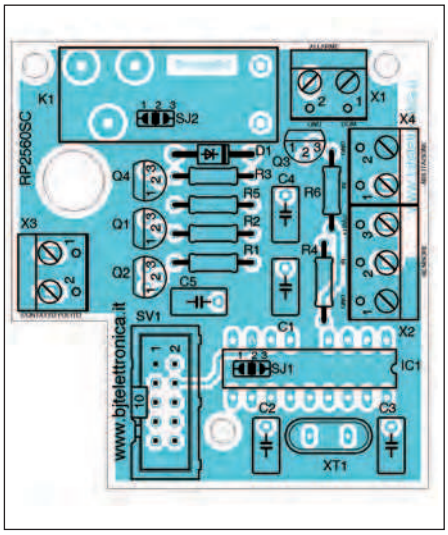


Figura 8: piano di montaggio scheda di controllo.

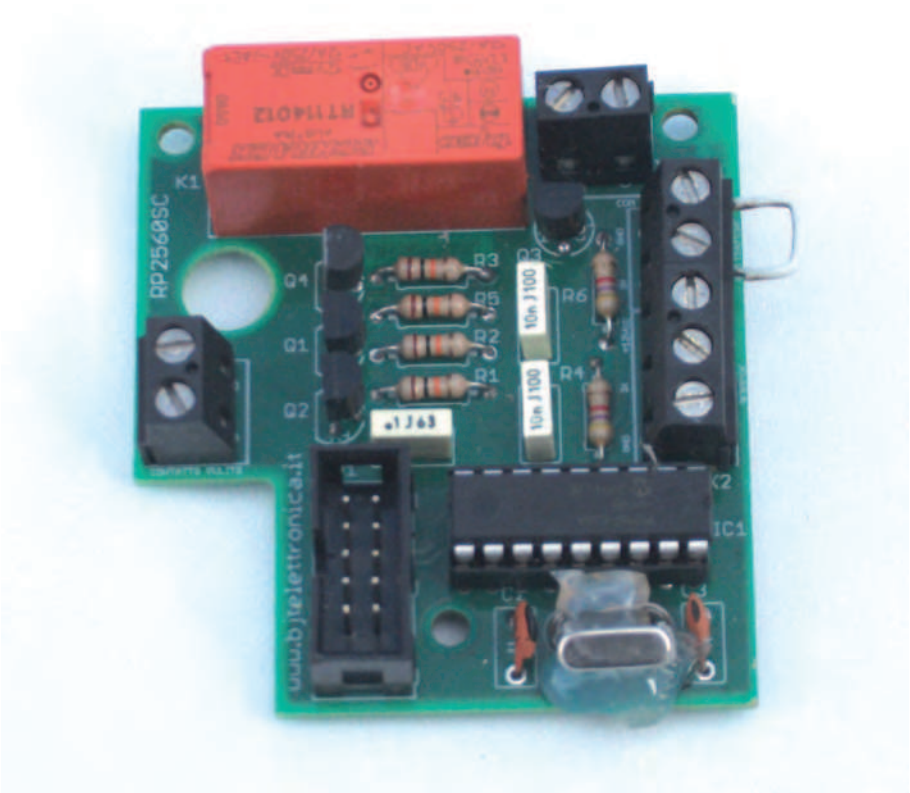


Figura 9: scheda di controllo assemblata.

Il pulsante S2 collegato al piedino 24 di IC1, ha la funzione di riportare la riproduzione all'inizio, in modalità di registrazione, se premuto riporta la memoria all'inizio in modo da poter registrare sulla precedente registrazione, in modalità di riproduzione riporta all'inizio della memoria pronta per essere riprodotta.
La resistenza R1 ha la funzione di pull up sul pulsante S1, mentre la resistenza R2 ha la funzione di pull down sul pulsante S2. Il diodo led LED1, collegato sul piedino 25 di IC1, viene acceso quando viene avviata la registrazione o la riproduzione.
Il piedino 24 di IC1 è collegato al piedino 5 del connettore SV1, il piedino 25 di IC1 connesso al piedino 4 del connettore SV1 e il piedino 23 di IC1 è collegato al piedino 6 del connettore SV1. Tramite SV1 è possibile collegare la scheda del registratore alla scheda di controllo che sarà trattata più avanti. Dai piedini 15 e 14 di IC1 fuoriesce il segnale amplificato prelevabile tramite il morsetto X3. A questo morsetto è possibile connettere un altoparlante da 8 con potenza massima di 1W, invece dalla presa jack X2 esce un segnale con cui è possibile pilotare delle casse acustiche da PC. La rete RC, composta dalla resistenza R6 e dal condensatore C5, ha la funzione di attenuare il segnale e di disaccoppiarlo da eventuali componenti continue. Il ponticello SJ1 se chiuso verso massa, il messaggio viene ripetuto all'infinito mentre se chiuso

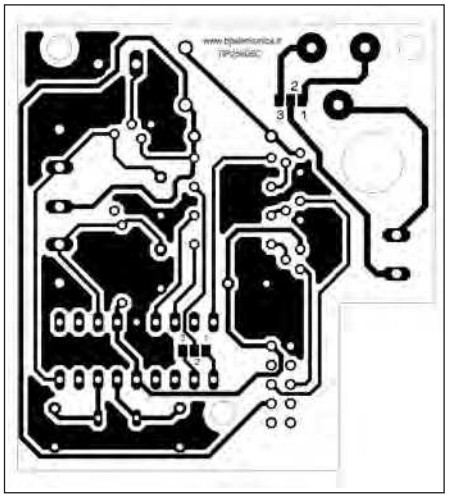


Figura 10: circuito stampato scheda di controllo in scala 1:1 (lato rame).

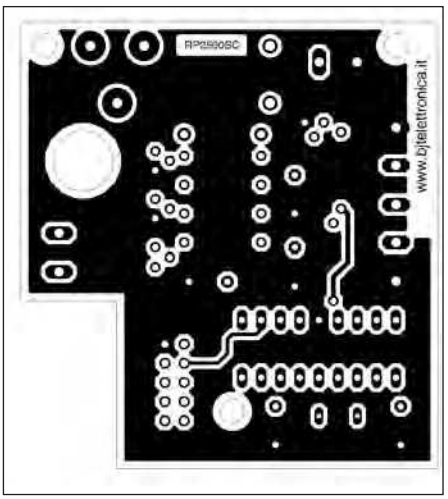


Figura 11: circuito stampato scheda di controllo in scala 1:1 (lato componenti).

verso +5V, il messaggio viene ripetuto una sola volta, questa è la posizione consigliata per il funzionamento di questo progetto. X1 è il morsetto da cui giunge la tensione d'alimentazione. Questa deve essere tra 9V e 12V stabilizzata, il diodo D1, posto immediatamente dopo il morsetto X1, ha la funzione di proteggere il circuito dalle inversioni di polarità. Da questo diodo si suddividono due rami: uno che va direttamente al piedino 8 del connettore SV1 che serve ad alimentare il relè e il sensore a doppia tecnologia sulla scheda di controllo. L'altro ramo che va allo stabilizzatore IC2 che ricava la tensione di alimentazione a 5V sia per l'integrato IC1 e per la scheda di controllo in quanto l'uscita di IC2 è collegata anche al piedino 2 del connettore SV1. La **figura 4** mostra lo schema elettrico della sche-

mikroC PRO for PIC

dsPIC

PIC32

everywhere

If you need more power, just switch over from PIC or dsPIC to PIC32 easily. We have been carefully planing backward compatibility for over 500 of our library functions, so you will be able to literally copy-paste your existing codes and build them with just a few adjustments.

CODICE MIP 2806166

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

www.mikroe.com

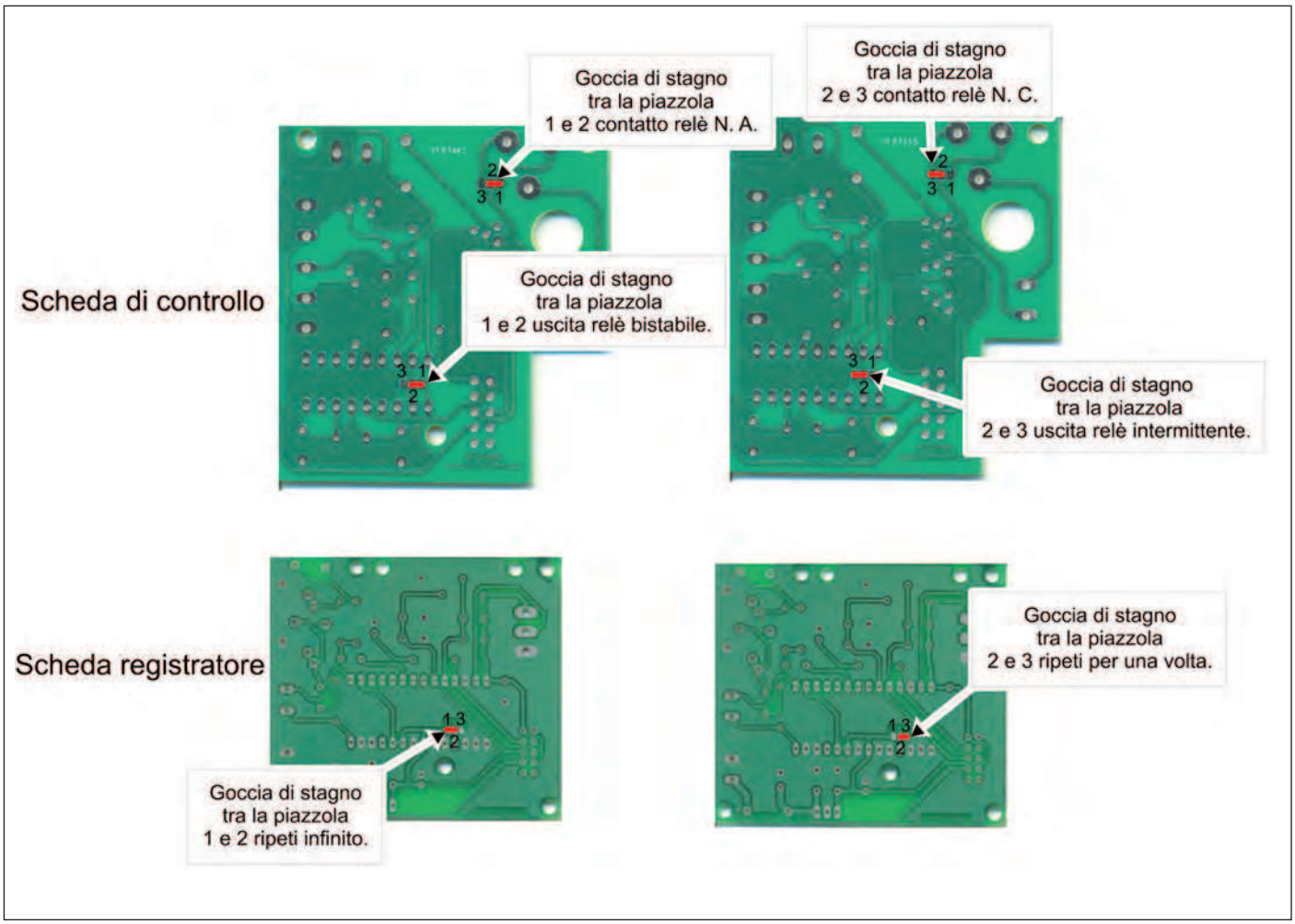


Figura 12: impostazione dei ponticelli a saldare.

da di controllo, il tutto e gestito dal microcontrollore IC1: un PIC16F628 che si occupa della parte operativa. La porta RB è stata utilizzata come porta di uscita. Mentre la porta RA come porta d'ingresso. Per questo progetto è stato usato un quarzo (XT1) esterno da 4 MHz e la modalità di RESET interna. Dalla porta RA4 di IC1 entra il segnale proveniente dal sensore a doppia tecnologia, prelevato tramite la morsettiera X2, il sensore al suo interno contiene un relè con contatto normalmente chiuso verso massa, quando il sensore viene attivato, apre il contatto del relè facendo sì che la porta RA4 si porti a potenziale +5V tramite la resistenza di pull up R4. Questa condizione porta all'attivazione del messaggio registrato sulla scheda del registratore, questo avviene tramite la porta RB0 che, portandosi a livello alto, porta in conduzione il transistor Q1, il quale attiva il registratore. Le uscite RB4 e RB5 di IC1 sono usate per pilotare il relè K1 tramite il transistor Q4 la cui base può essere collegata a una delle due uscite RB4 o RB5, tramite il ponticello SJ1. Pilotando Q4 con RB4 il relè sarà acceso permanentemente, mentre connettendo RB5 il relè verrà acceso ad intermittenza. Il relè K1 viene disattivato alla fine della riproduzione del messaggio audio con un piccolo ritardo. Sempre nello stesso momento, parte il conteggio di un timer programmato per 5 secondi entro questo tempo, se il sensore è ancora attivo, viene attivata la porta RB3, la quale portandosi a livello alto provoca la chiusura verso massa del transistor Q3 che attiva l'ingresso di allarme della centrale antifurto (morsetto X1). Alla fine della riproduzione del messaggio, viene attivata la porta RB1 che chiude il transistor Q2 a +5V, questo fa sì che il messaggio viene riportato all'inizio, pronto per essere riprodotto di nuovo. Il morsetto X4 consente di abilitare il dispositivo tramite centrale consentendo, quando la centrale è disattivata, lo spegnimento del dissuasore; effettuando un ponticello sul morsetto X4 si può utilizzare il dispositivo autonomamente alla centrale antifurto (vedi **figura 9**).

REALIZZAZIONE PRATICA

Per la realizzazione di questo progetto si deve utilizzare il master di **figura 7** per la realizzazione della scheda registratore, i master di **figura 10** e **11** per la realizzazione della scheda di controllo.
Mediante la tecnica della foto incisione, realizzate la basetta mono faccia del registratore e poi quella doppia faccia della scheda di controllo. In questa scheda

Guarda il video del dissuasore in funzione!

dovete realizzare i fori metallizzati con gli appositi rivetti. Montate le due schede seguendo la disposizione dei componenti di **figura 5** per la scheda registratore e di **figura 8** per la scheda di controllo.
In **tabella 1** è riportato l'elenco componenti per la scheda registratore, mentre in **tabella 2** l'elenco componenti della scheda di controllo.
Si raccomanda di montare per prima tutti componenti a basso profilo, resistenze, condensatori e zoccoli, passando poi ai transistor, connettori, morsetti, pulsanti. Si consiglia di utilizzare un saldatore da 30W a punta fine con la apposita spugnetta ben inumidita. Per la programmazione del PIC16F628, è necessario abilitare dal vostro software del programmatore l'oscillatore in modalità XT per quarzi o risuonatori con range da 200KHz a 4MHz, abilitare il MCLR modalità interna cioè il piedino MCLR diventa una porta di input e output che abilitata una rete di RESET interna, infine bisogna abilitare "Watch-Dog Timer", tutte le altre voci devono essere disabilitate. Eseguite i ponticelli SJ1 sulla scheda registratore fondendo una goccia di stagno sulle piazzole 2 e 3 (vedi **figura 12**), anche per la scheda di controllo dovete eseguire i ponticelli SJ1 e SJ2: scegliendo in base alle vostre esigenze il funzionamento del relè K1. Se il relè deve funzionare in modalità bistabile o intermittente, con contatto normalmente chiuso o aperto, sempre in **figura 12** sono illustrati i ponticelli da realizzare in base alla vostra scelta. Finita la fase di montaggio delle due schede (vedi **figura 6** e **9**), realizzate il cavo piatto di collegamento tra le due schede.

I sensori
A DOPPIA TECNOLOGIA

I sensori a doppia tecnologia abbinano due tecniche: quella a microonda e quella all'infrarosso. Questo permette di ridurre i falsi allarmi. La parte a microonda è costituita da due elementi (emettitore e ricevitore) che sono collocati nello stesso involucro. L'emettitore è costituito da un diodo Gunn. Applicando ad esso una determinata tensione, si genera un'altissima radiofrequenza, ossia microonde. Il ricevitore è, invece, un diodo Schottky. Entrambi i dispositivi, l'emettitore e il ricevitore, sono contenuti in un modulo denominato cavità. L'energia a microonde viene irradiata da un'antenna trasmittente in uno spazio chiamato campo, delimitato dalla forma dell'antenna costituita dalla cavità. Ad un'altra antenna, ricevente, è affidato il compito di captare la parte a microonde riflessa dagli oggetti in movimento. Infatti, se la posizione degli oggetti situati nel campo protetto rimane stazionaria, il segnale riflesso avrà la stessa frequenza di quello emesso. Se, però, uno degli oggetti si muove, la frequenza riflessa varierà, dando l'impulso di allarme. Questo fenomeno è noto come effetto Doppler. La parte ad infrarossi, invece, impiega un sensore di tipo termoelettrico che controlla, attraverso un sistema ottico, la zona protetta. Il sistema a infrarosso è costituito da un ricevitore in quanto l'emettitore si identifica con il calore irradiato dalla persona che si muove nell'ambiente. Ogni corpo animato o inanimato avente una temperatura superiore allo zero assoluto emette radiazioni calorifiche, la cui lunghezza d'onda si colloca nel campo dei raggi infrarossi. L'emissione termica è proporzionale alla temperatura, alla massa del corpo stesso. Il sensore rimane però insensibile a lente variazioni di livello delle radiazioni stesse, come quelle che si verificano negli impianti di riscaldamento, all'alternasi del giorno e della notte o del tempo soleggiato o coperto. Il sensore termoelettrico è munito di un sistema ottico che suddivide l'area in una serie di zone. Alcune sono sensibili alla radiazione infrarossa, altre non lo sono. Una persona che si muove da una zona sensibile ad una non sensibile (o viceversa), provoca una variazione nell'energia infrarossa che viene rilevata dal sensore. Un amplificatore rileva le variazione del segnale che poi viene tradotto in un segnale di allarme. I segnali di allarme, provenienti dalle due tecnologie, vengo poi elaborati da un microprocessore che effettua l'operazione di AND logico. Il sensore ecciterà il proprio relè soltanto se saranno attivate entrambe le due tecnologie, questo permette di limitare i falsi allarmi che si possono presentare utilizzando solo una tecnologia. In questo progetto è stato usato un sensore a doppia tecnologia di produzione AMC Elettronica (vedi figura 16) esso dispone di un dip switch con cui è possibile selezionare oltre all'operazione AND anche l'operazione OR che aumenta la sensibilità del sensore.

Figura 16: sensore a doppia tecnologia Mouse 07 di AMC Elettronica.

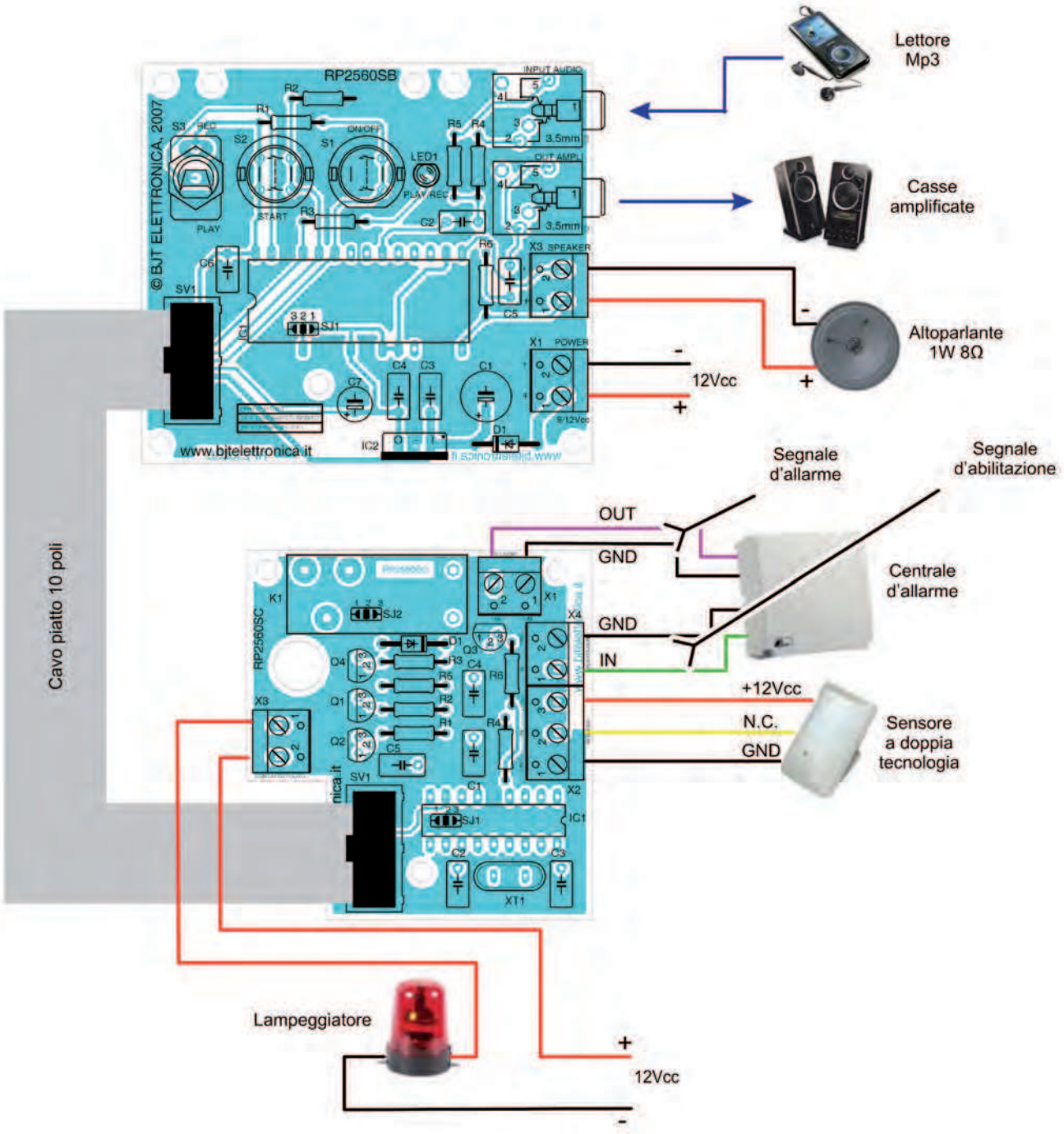


Figura 13: schema di collegamento delle due schede e ai vari componenti.

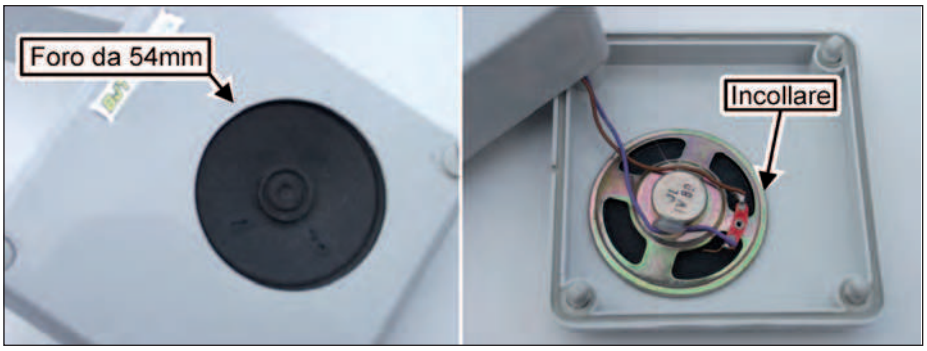


Figura 14: particolari dell'assemblaggio dell'altoparlante.

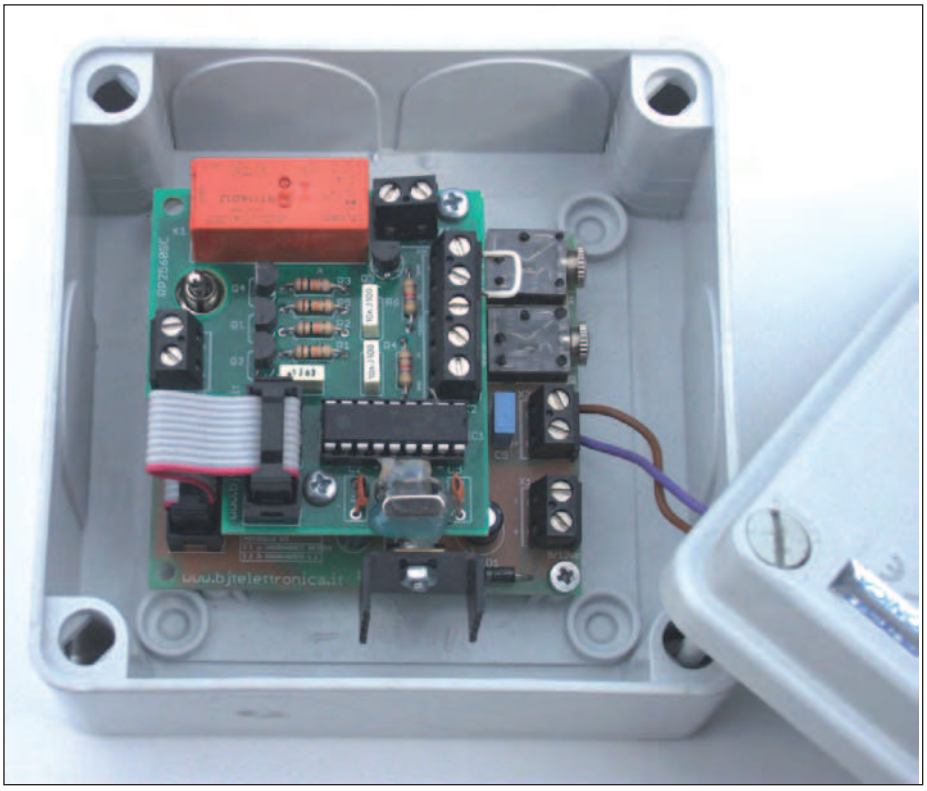


Figura 15: le due schede assemblate dentro il contenitore.

Osservando la **figura 13** disporre i connettori IDC a 10 poli sul cavo piatto, cioè con le tacche rivolte verso l'interno del cavo. Premere con forza i connettori su un tavolo affinché si perfori l'isolante e il connettore non si muova più sul cavo.
Ora passiamo alla registrazione di un messaggio sulla scheda registratore: la prima operazione da fare è alimentare la scheda con una tensione di 12V. Connettere un lettore di MP3 alla presa jack X4 con l'apposito cavo con due spine jack da 3,5mm (vedi **figura 13**) e con il messaggio registrato, spostare la levetta del deviatore S3 in posizione Rec. Premere il tasto S2 (Start), mettete in play il lettore di MP3 e premete il tasto S1 (ON/OFF): la scheda inizierà la registrazione. Alla fine della registrazione, premere nuovamente il tasto S1, cosi facendo abbiamo finito la fase di registrazione.
Per ascoltare la registrazione, collegare alla scheda registratore un piccolo altoparlante di 1W come mostrato in **figura 13**, mettere il deviatore S3 in posizione Play, premere il tasto S1 (ON/OFF): inizierà la riproduzione del nostro messaggio. Premendo di nuovo il tasto S1 (ON/OFF), la riproduzione si metterà in pausa, premendo il tasto S2 (Start) il messaggio verrà riprodotto dall'inizio. Per eseguire il collaudo della scheda di controllo, bisogna effettuare i collegamenti come mostra la **figura 13**. Non collegare la centrale antifurto alla scheda, ma è necessario effettuare un ponticello sul morsetto di abilitazione X4 come mostra **figura 15**. Collegando solo il sensore a doppia tecnologia, alimentando le schede e passando davanti al sensore, dovremmo sentire il nostro messaggio e udiremo eccitarsi il relè K1. Se la potenza sonora ottenuta dall'altoparlante da 1W non vi soddisfa, potete connettere delle casse da PC all'uscita jack X2, migliorando la resa acustica. Verificato che tutto funzioni, si può passare all'assemblaggio nel contenitore.

ASSEMBLAGGIO NEL CONTENITORE

Il contenitore usato per il dissuasore audio è una scatola per derivazione elettriche a pareti lisce (quello usato in questo progetto è di marca ABB) dalle dimensioni 100x100mm. L'unica operazione da fare, è realizzare un foro da 54mm sul coperchio per posizionare correttamente il piccolo altoparlante da 1W (vedi **figura 14**). Poi passate al fissaggio dell'altoparlante, passando della colla ciano acrilica sui bordi dell'altoparlante, appoggiate dietro al coperchio, come mostra **figura 14**, e lasciate asciugare bene.
Montate le due schede, come mostra la **figura 15**, cioè una sopra l'altra utilizzando gli appositi distanziatori,
fissate la scheda registratore al contenitore con delle viti autofilettanti 6x3mm sulle apposite asole del contenitore.
Con questa operazione si è conclusa la realizzazione del nostro dissuasore audio (vedi **figura 1**), non rimane che trovare una sistemazione adeguata per metterlo in funzione.

CONCLUSIONE

Quanto esposto in questo articolo, permette di fare chiarezza sull'utilizzo del dissuasore audio e del circuito integrato ISD2560 in modo da trarre spunto per i vostri futuri progetti "parlanti".
Il firmware e tutti file per la realizzazione di questo progetto, sono disponibile per il download dal sito di Fare Elettronica. ❑

CODICE MIP 2804883

di GRAZIA ANCONA

filtri ATTIVI con amplificatori operazionali (parte terza)

La trattazione continua in questa terza ed ultima puntata affrontando due argomenti molto importanti: i filtri selettivi ed i filtri elimina banda (notch)

Nella prima puntata di questa serie di articoli, è stato già introdotto il concetto di filtro selettivo. Sostanzialmente si tratta di una rete filtrante che amplifica un gruppo di frequenze comprese fra due valori di frequenza di taglio. Questi tipi di filtri sono usati ove è necessario far transitare un insieme di frequenze molto ristretto come, ad esempio, negli analizzatori di spettro audio. Altre applicazioni possono essere: sistemi di luci psichedeliche e generazione di suoni elettronici. Questi filtri sono definiti a banda stretta in quanto il gruppo di frequenze che transitano nella loro banda passante non è mai elevato e di solito non supera due volte il valore della frequenza centrale. Per maggior chiarezza se la relazione:

$$B \geq 2\ fo$$

è soddisfatta occorre un filtro selettivo a banda larga. Nella prima puntata sono stati anche dettagliati i parametri che ne interessano la curva di risposta di questo tipo di filtri e, di conseguenza, la loro funzione di trasferimento. Proviamo a ricordarli:

- B: banda passante del filtro, ovvero la gamma di frequenze che devono transitare amplificate (oppure non attenuate). B è compreso fra le due frequenze di taglio (inferiore e superiore).
- fo: frequenza centrale del filtro. A questa frequenza l'amplificazione è massima, alcuni testi indicano questa frequenza con il nome di risonanza.
- ftl: frequenza di taglio inferiore.
- fth: frequenza di taglio superiore.
- Q: grado di pendenza del filtro (in stretta relazione con B).

Il Q, che può anche essere definito il fattore di merito del filtro, è un parametro molto importante nella stesura di un progetto con filtri selettivi e merita di essere approfondito adeguatamente. La **figura 1** mostra graficamente il concetto di Q che, ricordiamolo, è un numero puro. Osservando la **figura 1**, possiamo notare come le pendenze dei filtri aventi differente valore di Q tendano sempre ad approssimarsi alla pendenza di 20 dB per decade (6 dB per ottava) quando la frequenza si allontana dal valore centrale (fo). Tutto ciò accade per qualsiasi larghezza di banda (B) e per valori delle frequenze di taglio (ftl e fth) alquanto differenti. Il circuito più comunemente usato per realizzare filtri selettivi a banda stretta è il filtro passa banda a reazioni multiple. Viene definito così in quanto il circuito che ripropone parte del segnale d'uscita all'ingresso (controreazione) è composto da una parte resistiva e da una capacitiva. Non è la sola tipologia circuitale possibile ma, a giudizio di molti, è un ottimo compromesso fra prestazioni e semplicità progettuale, per cui verrà presentata in queste pagine. In **figura 2 a** vi è la soluzione circuitale per alimentazione singola mentre, in **figura 2 b**, possiamo trovare la schematizzazione

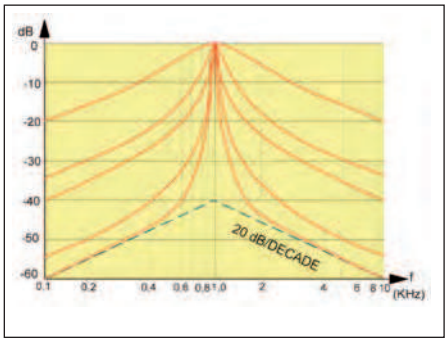
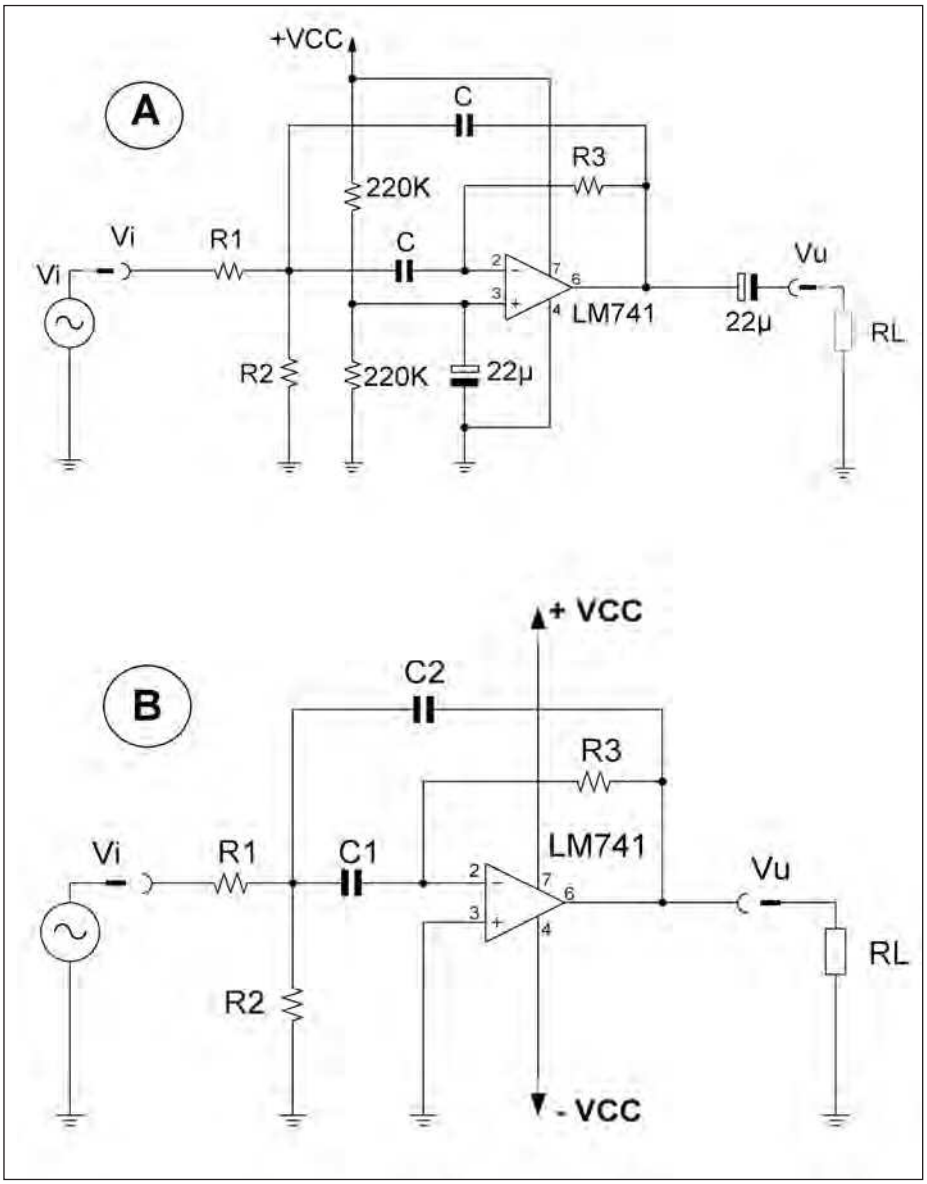


Figura 1: differenti risposte di filtri selettivi per diversi valori di Q.



per quella con alimentazione duale. Un inconveniente di questa tipologia circuitale è costituito dal fatto che il guadagno in banda passante ed il Q sono legati fra loro. In particolare, aumentando l'uno aumenta anche l'altro. Per semplificare i calcoli di progetto è opportuno che le due capacità siano di identico valore. Diciamo subito che, per evitare auto-oscillazioni e risposte imprevedibili nella funzione di trasferimento, è bene che il Q non superi le 20 -22 unità. Le relazioni che permettono di impostare un progetto con un filtro passa banda sono mostrate in **tabella 1**. Nella **tabella 1** la massima amplificazione in banda passante è preceduta da un segno negativo. Questo non vuol dire che il filtro attenua bensì che la fase del segnale in uscita è spostata di 180° rispetto al segnale d'ingresso. Ciò è dovuto all'applicazione del segnale da filtrare sul terminale d'ingresso negativo dell'amplificatore operazionale (il terminale d'ingresso positivo è collegato a massa direttamente). Al lettore attento non sarà sfuggito che in questo tipo di circuito il segnale di controreazione viene confrontato direttamente con il segnale d'ingresso. Il circuito proposto ha un'interessante particolarità: è possibile cambiare la frequenza centrale (detta anche frequenza di risonanza) semplicemente cambiando il valore di una resistenza, ovvero la no-

Figura 2: filtro selettivo a reazione multipla con alimentazione singola (A) e duale (B).

TABELLA 1

FORMULE DI PROGETTO PER FILTRO SELETTIVO A REAZIONE MULTIPLA

Parametro da calcolare	Formula	Definizione
fo≈	$= (\sqrt{(1 / (R3\ C^2))}\ (1 / R1 + 1/ R2)) / 2\pi$	Frequenza centrale
Q	$= (2\pi\ fo\ R3\ C^2) / 2\ C$	Fattore di merito
B	$= fo / Q$	Banda passante
fth	$\approx fo + (B / 2)$	Frequenza di taglio superiore (per $Q \geq 5$)
ftl	$\approx fo - (B / 2)$	Frequenza di taglio inferiore (per $Q \geq 5$)
A	$= - R3 / 2\ R1$	Amplificazione massima (alla frequenza centrale)
R1	$= Q / (A\ \ C\ \ 2\pi\ fo)$	-
R2	$= Q / ((2Q^2 - A)\ C\ 2\pi\ fo)$	-
R3	$= 2\ R1 / A$	-

stra R3. Infatti, vale la relazione (in cui il carattere " ' " indica i valori della nuova frequenza di risonanza):

$$R3' = R3\,(fo / fo')^2$$

Inoltre, è possibile ricavare il valore di Q anche avendo disponibili i valori delle frequenze di taglio e di risonanza:

$$Q = fo / (fth - ftl)$$

Il tipo di filtro presentato è definito di secondo ordine. In realtà tale definizione potrebbe sembrare un po' fittizia in quanto, osservando nuovamente la **figura 1**, possiamo notare che per valori di frequenza lontani dalla risonanza, la pendenza del filtro è 20 dB/decade o 6 dB/ottava. I lettori ricorderanno che tali valori erano propri dei filtri di primo ordine VCVS (vedi puntate precedenti). Tuttavia l'ordine di un filtro è deciso anche da particolari elementi della sua funzione di trasferimento, in particolare dai cosiddetti poli. Senza troppo approfondire, diremo che i poli di una funzione sono quei particolari numeri capaci di mandare la funzione stessa ad un valore numerico infinito. I filtri a reazioni multiple hanno un polo nella loro funzione di trasferimento e, per definizione, un singolo polo contraddistingue una cella di ordine secondo. Questo discorso serve ad introdurre la possibilità di ottenere filtri aventi pendenze fuori risonanza più ripide, in particolare del quarto, del sesto e dell'ottavo ordine **(vedi figura 3)**. Per ottenere filtri passa banda di ordine superiore al secondo, come si evince dalla **figura 3**, è

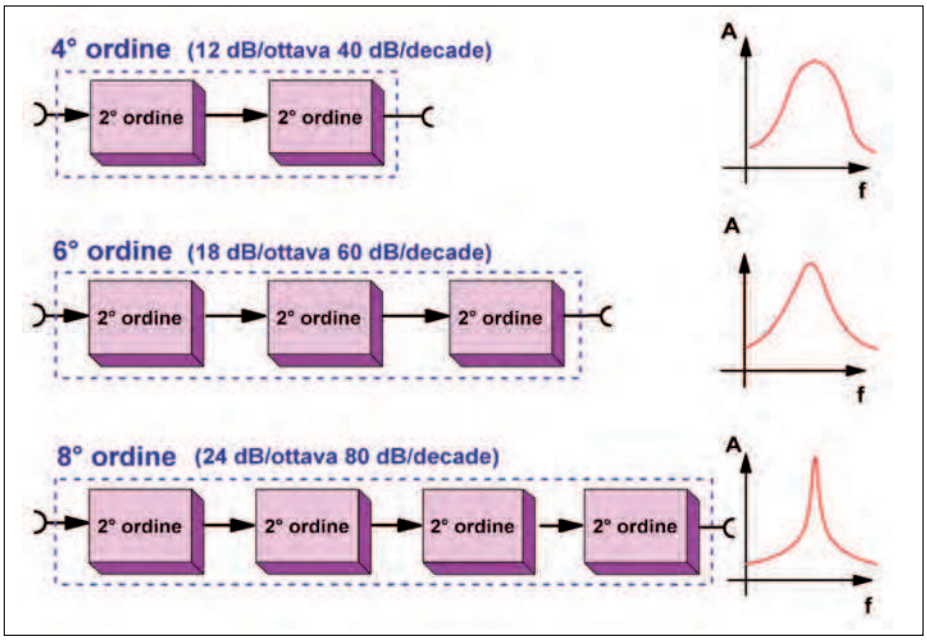


Figura 3: filtri di ordine secondo posti in cascata per aumentare il Q complessivo.

Frequenze fondamentali e loro armoniche

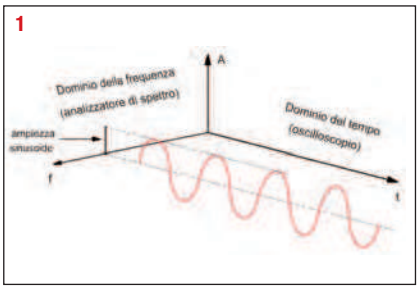
Noi siamo abituati a considerare un segnale alternato definendo un ampiezza e la sua frequenza. Faccio un esempio. La rete luce, in Europa, è fornita dai vari gestori con una tensione pari a 230 VCA e con frequenza di 50 Hz. Se questo è vero per la maggior parte dei problemi che riguardano la nostra vita quotidiana, come ad esempio usare un asciugacapelli, vi sono casi in cui la questa affermazione deve essere un po' approfondita. In effetti un segnale sinusoidale perfetto, ovvero privo di alcun tipo di distorsione, può essere effettivamente pensato come composto da una sola frequenza, quella che viene definita frequenza fondamentale o semplicemente "fondamentale". Generalmente una funzione sinusoidale viene raffigurata in due modalità, meglio definite come "domini", che sono:

- Dominio del tempo

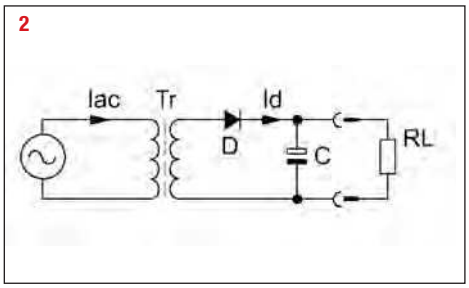
- Dominio della frequenza

*Nel primo caso osserveremo, magari con l'aiuto di un oscilloscopio, la classica forma che ben conosciamo. Nel secondo invece avremo a che fare con un segmento verticale e per visualizzare ciò ci occorrerà un analizzatore di spettro **(vedi figura 1)**. Purtroppo, nella realtà, di funzioni sinusoidali prive di distorsione ve ne sono ben poche. Infatti, è sufficiente applicare alla forma d'onda fornita dal gestore un qualsiasi carico non perfettamente resisitivo per deformare, in maniera più o meno evidente, la nostra forma d'onda così perfetta. Si osservi il banale circuito d'alimentazione presentato in **figura 2**. Supponendo che la forma d'onda di*

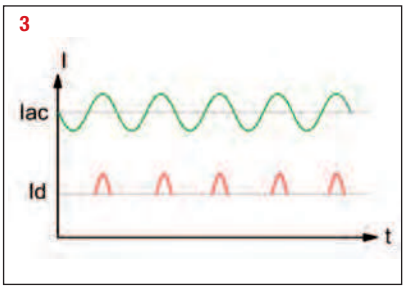
1



2



3



sufficiente collegare in cascata più celle. Occorre, tuttavia, tenere presenti alcune limitazioni:
Possiamo ottenere solo ordini totali pari (una singola cella possiede ordine secondo).
Per evitare instabilità non bisogna andare oltre l'ottavo ordine (4 celle di secondo ordine). Meglio sarebbe non superare le 3 celle totali (sesto ordine).
Più aumenta l'ordine e meno dovrebbe essere il guadagno alla frequenza centrale (per evitare instabilità).
Tutte le celle devono possedere uguale guadagno (A) e fattore di merito (Q). In altre parole devono essere identiche.
Inoltre, il Q totale della catena filtrante non sarà il semplice prodotto del Q delle varie celle ma subirà una variazione decrementale in ragione del numero delle celle stesse. La variazione del Q totale è ricavata da un coefficiente come mostrato in **tabella 2**. Facciamo un piccolo esempio di come usare la **tabella 2**. Per prima cosa, occorre considerare l'ordine del filtro risultante, esso sarà la nostra prima specifica di progetto. Supponiamo che sia sufficiente un Q pari a 10.

Adesso occorre definire la pendenza del filtro, essa costituirà la nostra seconda specifica di progetto. Per la nostra ipotetica applicazione supponiamo che ci occorra una pendenza pari a 60 dB/decade (18 dB/ottava). Con questi parametri la nostra cella filtrante può rientrare nella categoria dei filtri abbastanza selettivi, adatti cioè ad amplificare un campo di frequenze molto ristretto. Bene, per ottenere una risposta di 60 dB/decade occorreranno 3 celle di secondo ordine identiche (vedi **figura 3**). Il problema che

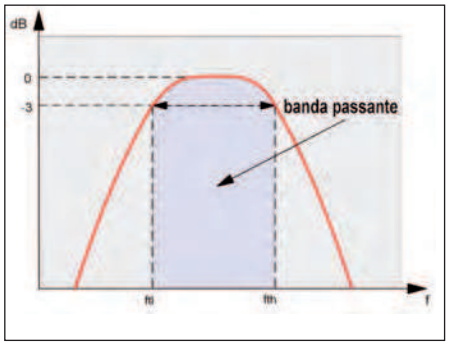


Figura 4: diagramma ampiezza-frequenza di un generico filtro selettivo.

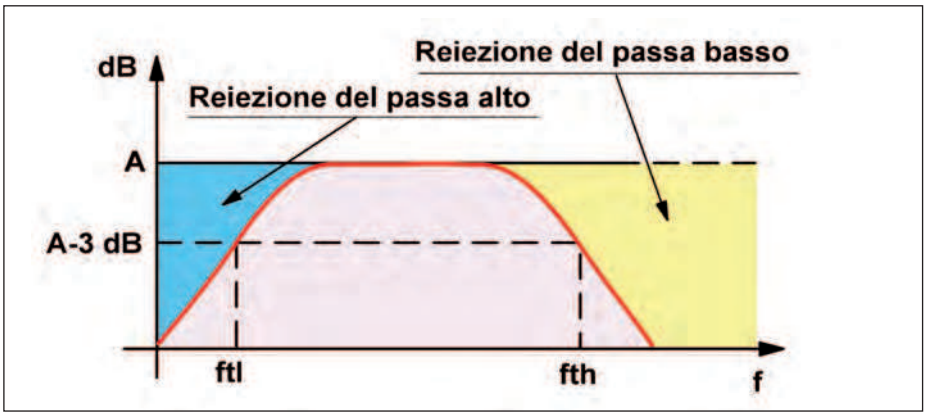
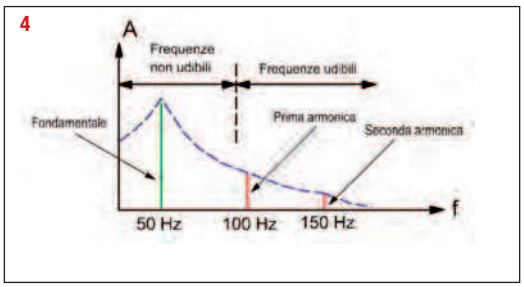


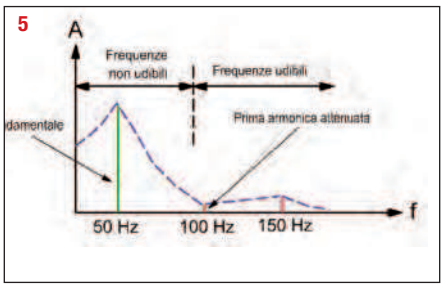
Figura 5: metodo per ottenere un filtro passa banda.

*corrente applicata la primario del trasformatore sia pressoché perfetta - ovvero priva di distorsioni - non possiamo dire altrettanto per la corrente I che scorre nel diodo **(vedi figura 3)**. Infatti la natura della corrente circolante, per motivi legati alla configurazione circuitale ed alla presenza del condensatore C, ci presenta una forma d'onda che sinusoidale proprio non è! E' noto da tempo che una frequenza fondamentale è accompagnata da un certo numero di frequenze associate, dette armoniche, che possiedono ampiezza decrescente e frequenza multipla della fondamentale. Ad esempio, per la tensione di rete si avrà:*

Frequenza fondamentale = 50 Hz
Prima armonica = 100 Hz
Seconda armonica = 150 Hz
n armonica = n x 50 Hz

L'ampiezza ed il numero significativo di queste frequenze armoniche è proporzionale al grado di distorsione, rispetto ad una funzione sinusoidale perfetta, della frequenza fondamentale. Al limite, una forma d'onda rettangolare (onda quadra) avrà un numero infinito di armoniche.
*Osserviamo adesso un grafico nel dominio della frequenza ove, ad una fondamentale di 50 Hz, sono state associate delle frequenze armoniche di ampiezza decrescente **(vedi figura 4)**. Come si noterà la fondamentale (50 Hz) non rientra nelle frequenze udibili. Non altrettanto si può dire della prima armonica che, essendo a 100 Hz, potrebbe essere udita anche se la sua ampiezza è minore rispetto alla fondamentale.*
*L'inserzione di un filtro notch centrato a 100 Hz potrebbe abbassare l'ampiezza della prima armonica così come mostrato indicativamente in **figura 5**.*
Lo studio delle armoniche può essere affrontato utilizzando la cosiddetta analisi di Fourier (studioso francese che visse a cavallo fra il 700 e l'800).





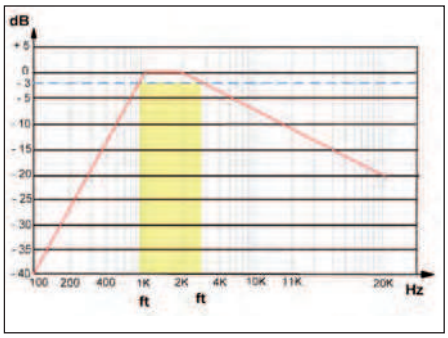


Figura 6: filtro passa banda con risposta asimmetrica.

si pone è conoscere il Q di una singola cella per ottenere il Q complessivo desiderato, ovvero 10. Consultando la **tabella 2**, troviamo il coefficiente 0,510 relativo ad una rete filtrante con pendenza pari a 18 dB/ottava. Esso sarà il nostro "numero magico", infatti:

$$10 \times 0{,}510 = 5{,}1$$

Dove il valore ricavato (5,1) sarà il Q di ogni singola cella. Come si noterà, esso è decisamente superiore al valore calcolato dividendo il Q totale per il numero delle celle. Infatti:

$$10 / 3 = 3{,}33$$

In altri termini, disporre di più celle in cascata comporta un decremento del Q di ogni singola cella e questo fenomeno è tanto più evidente quanto più è alto il numero delle celle costituenti la rete filtrante.

FILTRI SELETTIVI A BANDA LARGA

I filtri esaminati finora sono contraddistinti da una banda passante molto stretta, decisamente non adatta alle applicazioni ove deve transitare un gruppo abbastanza ampio di frequenze. Si immagini, ad esempio, un sistema di amplificazione destinato alla telefonia. Com'è noto, la banda passante telefonica è racchiusa nel campo di frequenza che va da 300 Hz a 3,4 KHz, ovvero una larghezza di banda pari a 3,1 KHz. Usare in questo caso un filtro selettivo a banda stretta implicherebbe due problematiche. Vediamole:

- Usare Q molto piccoli e, quindi, non avere una reiezione delle frequenze fuori banda sufficiente.
- Usare Q sufficienti limitando fortemente la banda passante rispetto ai valori previsti.

Queste circostanze possono evitarsi con l'uso di appropriati filtri selettivi definiti, per l'appunto, a banda larga. In **figura 4** appare la rappresentazione grafica della funzione di trasferimento di un filtro a banda larga generico. Le applicazioni dei filtri a banda larga sono molteplici. Della telefonia si è già parlato, tuttavia essi trovano uso ogni qualvolta è necessario limitare la banda passante del segnale in ingresso in modo da adattarla ai valori che ci interessano. Ad esempio, si consideri un sistema di amplificazione per sub-woofer (vedi **figura 5**). Questi amplificatori funzionano con potenze piuttosto elevate (fino ad 1 KW e anche più) ma con bande passanti tipicamente ridotte, di solito comprese fra 50 e 150 Hz. L'ingresso di questi amplificatori non può essere interessato dal segnale audio di tipo HI-FI poiché esso ha un campo di frequenza decisamente più vasto: circa 20 KHz di banda passante! E' quindi necessario anteporre un filtro che riduca decisamente questa eccessiva larghezza di banda, per l'appunto un filtro selettivo avente una banda passante di circa 60 ÷ 70

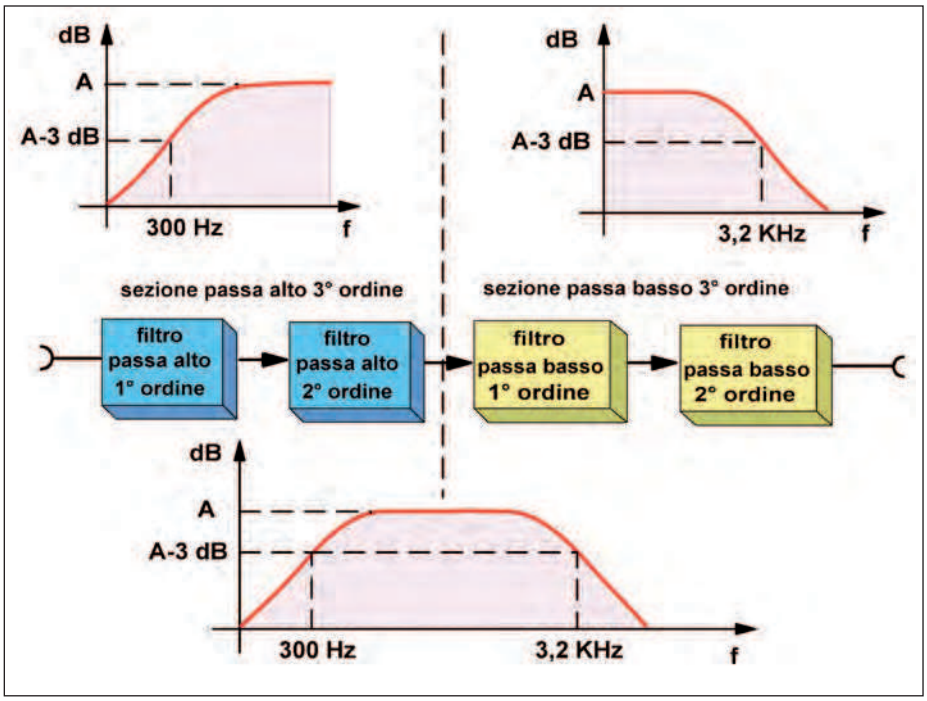


Figura 7: filtro passa banda di terzo ordine composto da due sezioni (passa alto e passa basso).

TABELLA 2

COEFFICIENTI PER LA COMPOSIZIONE DI FILTRI PASSA BANDA

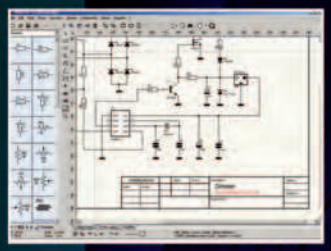
Numero di celle di 2° ordine	Filtro risultante	Pendenza in banda di reiezione	Coefficiente
2	4° ordine (2 poli)	12 dB/ottava (40 dB/decade)	0,644
3	6° ordine (3 poli)	18 dB/ottava (60 dB/decade)	0,510
4	8° ordine (4 poli)	24 dB/ottava (80 dB/decade)	0,435

Progettazione elettronica

facile per tutti!

sPlan 7.0

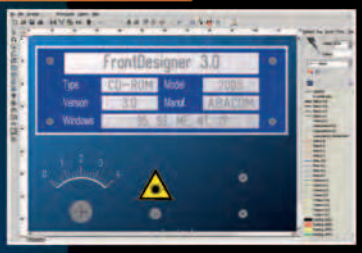
Programma per disegnare gli schemi elettrici.



Solo € 47,88

Front Designer

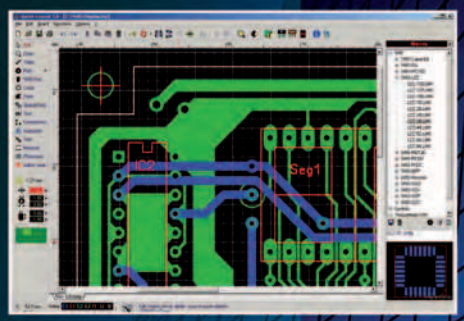
Programma per disegnare i pannelli frontali



Solo € 47,88

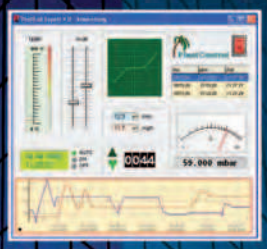
Sprint-Layout 5.0

Con Sprint-Layout è possibile disegnare il proprio PCB in modo facile e veloce.



Solo € 47,88

ProfiLab-Expert 4.0



Il software ProfiLab-Expert permette di sviluppare i propri progetti di misurazione digitale o analogica.

Solo € 119,88

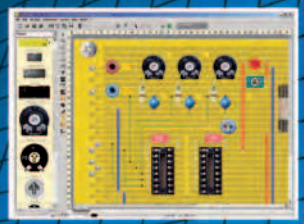
AudioWave 2.0

Con AudioWave 2.0 la propria scheda audio diventa un confortevole generatore di segnale LF che genera segnali da 1Hz a 20 kHz con una risoluzione di 1Hz.

Solo € 35,88

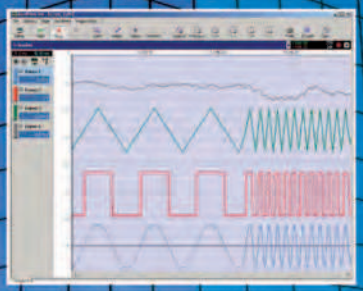
LochMaster 3.0

LochMaster 3.0 è uno strumento di sviluppo per la progettazione di boards. Possiede utili funzionalità per disegnare, documentare e testare una board.



Solo € 47,88

Real View 3.0



RealView è un'applicazione di misurazione che registra e traccia i valori acquisiti da dispositivi hardware analogici in tempo reale. Diverse curve possono essere tracciate in un singolo diagramma.

Solo € 47,88

Prezzi IVA inclusa

PUOI ORDINARE I SOFTWARE ABACOM SUL SITO www.ieshop.it/abacom
oppure telefonando allo 02 66504755

CODICE MIP 2808433

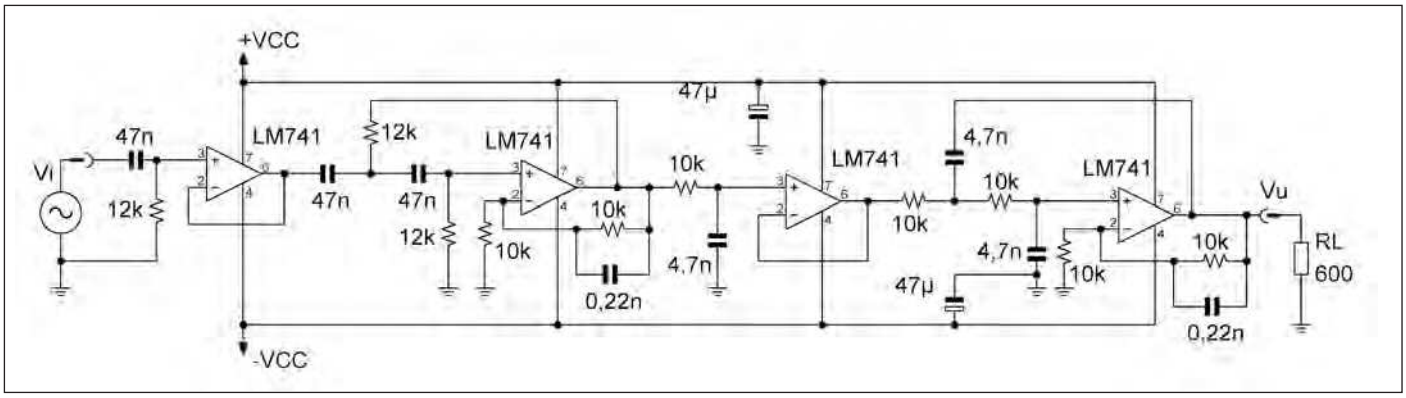


Figura 8: filtro passa banda di terzo ordine (schema elettrico completo).

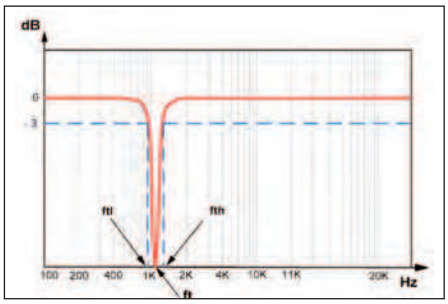


Figura 9: curva di risposta di un filtro notch a banda stretta.

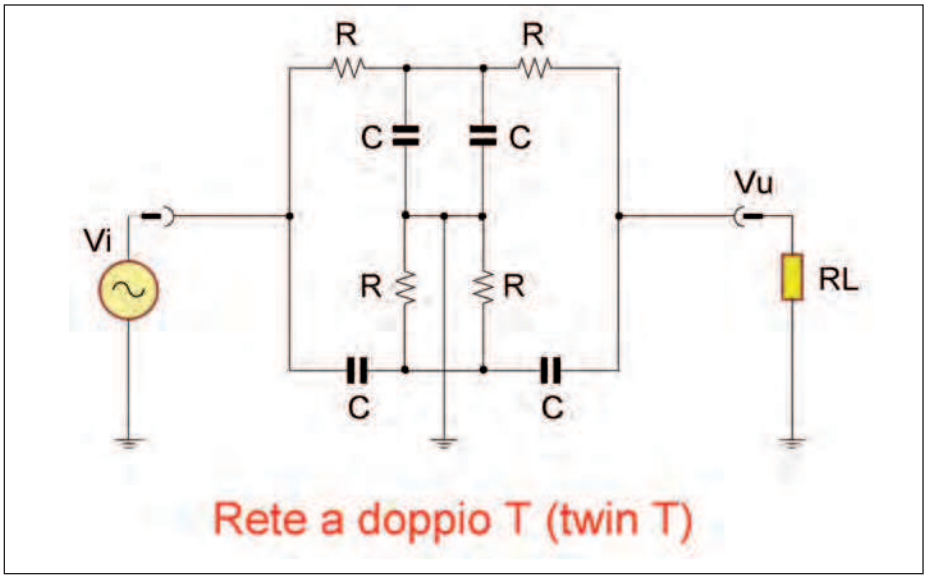


Figura 10: rete di controreazione definita a "doppio T"

Hz. Un'altra applicazione, decisamente più "seria", si ha nel campo biomedico. In questo settore si ha spesso a che fare con segnali generati dal corpo umano che, di solito, hanno frequenze da 10 ÷ 12 Hz fino a 200 ÷ 250 Hz. Tutto il resto rappresenta quello che in termini tecnici viene indicato con la parola generica di "rumore" e deve, quindi, essere decisamente eliminato. Anche in questo caso possono tornare utili i filtri selettivi. L'uso di questi ultimi è usuale anche in sistemi digitali ove è necessario convertire un segnale analogico (ad esempio, la musica) in un segnale discreto nel tempo.

Questo processo, definito conversione A/D (analogico -digitale), presuppone che venga fatto un drastico taglio delle frequenze che, per "naturale" limitazione dei componenti e soluzioni adottate, non possono essere convertite.

L'uso dei filtri selettivi è così spinto che, da alcuni anni a questa parte, sono stati immessi sul mercato speciali chip programmabili chiamati DSP (Digital Signal Processing) che svolgono le funzioni un tempo riservate all'elettronica analogica. L'uso di questi dispositivi consente elevate prestazioni ed elasticità dì uso non raggiungibili con i circuiti visti finora. Tuttavia, prima di affrontare la difficoltà dei DSP, è bene che il lettore disponga di pratica sui filtri analogici. Con le nozioni apprese fin qui è comunque possibile progettare filtri a banda larga anche di sesto ordine.

Vediamo come. Il metodo più comunemente usato è quello di disporre in cascata un filtro passa alto seguito da uno passa basso (vedi **figura 5**).

E' interessante notare che la ftl sarà la frequenza di taglio del filtro passa alto mentre la fth sarà quella del filtro passa basso. Occorre tenere presente che: l'ordine complessivo del filtro sarà identico a quello dei filtri usati. In linea teorica, ciascun filtro potrebbe avere diversa pendenza (ordine). In questo caso la risposta complessiva sarebbe asimmetrica (vedi **figura 6**). Sempre in linea teorica, il guadagno in banda passante potrebbe essere diverso nei due filtri. In questo caso ci troveremmo di fronte ad una risposta asimmetrica in ampiezza che potrebbe non essere di tipo Butterworth.

Il guadagno in banda passante, espresso in dB, sarà la somma dei singoli guadagni di ogni cella.

Facciamo un esempio. Si supponga di dover progettare un filtro passa panda con frequenza di taglio inferiore pari a 300 Hz e superiore di 3,2 KHz. Si vuole ottenere una pendenza fuori banda del terzo ordine Butterworth e ciò implicherà che il guadagno in banda passante sarà fissato a priori (non dimensionabile in fase di progetto). Il filtro sarà posto a valle di un generatore che fornisce un segnale di 1 Vrms e dovrà pilotare un carico pari a 600 Ω, sarà possibile usare un'alimentazione duale. Queste le specifiche. Innanzitutto, vediamo se è il caso di usa-

re un filtro a banda larga o se sia sufficiente un filtro a reazione multipla. Calcoliamo la frequenza centrale che, come si ricorderà, è la media geometrica delle frequenze di taglio:

fo = √(300 x 3200) = 980 Hz (arrotondato)

A fronte di una larghezza di banda pari a:

B = 3200 - 300 = 2900

Abbiamo adesso tutti gli elementi per decidere quale sarà l'architettura del nostro filtro. Infatti, introducendo l'argomento dei filtri selettivi, abbiamo premesso che un rapporto inferiore a 2 fra banda passante e frequenza centrale esclude l'impiego di un filtro a banda stretta. Nel nostro caso:

2900 / 980 = 2,9 (circa)

Valore, quest'ultimo, decisamente appropriato all'uso di un filtro a banda larga. Per ottenere una pendenza del terzo ordine di Butterworth sarà necessario utilizzare 2 celle per il passa basso ed altrettante per il passa alto.
Ogni sottoinsieme sarà composto da un filtro di primo ordine posto in cascata con un filtro di secondo ordine. Entrambi forniranno insieme una funzione passa banda come mostrato in **figura 7**. Il guadagno in banda passante di ogni sottosezione, come già accennato, è fissato a priori poiché è richiesta una risposta di tipo Butterworth. E' per noi necessario conoscere questo valore in quanto il segnale d'uscita non deve subire distorsioni da clipping (saturazione). Come si ricorderà (vedi seconda puntata) ciò potrebbe accadere se il segnale del generatore è troppo ampio rispetto al guadagno complessivo o se la tensione di alimentazione, fissata una volta per tutte dal progettista, non è sufficiente.
A questo punto occorre dare un'occhiata alla tabella 6 della seconda puntata. Essa riporta il guadagno per ogni ordine di filtro nella modalità Butterworh, sia in dB che in valore assoluto (ovvero Vu / Vi).
Per i calcoli attuali ci serviremo di quest'ultimo.
Nel nostro esempio vi sono due sottosezioni di 3° ordine che prevedono, singolarmente, un guadagno pari a 2. Da cui, in banda passante, il guadagno complessivo in valore assoluto sarà il prodotto del guadagno delle singole sottosezioni:

Atotale = Apa x Apb = 2 x 2 = 4

Mentre, usando i dB, avremmo avuto:

dBtotale = dBpa + dBpb = 6 + 6 = 12 dB

Sapendo che 1 Vrms corrisponde a 2,82 Vpp possiamo calcolare facilmente il valore minimo della nostra tensione di alimentazione che dovrà essere superiore al valore di cresta della tensione d'uscita. Quanto a quest'ultima, viste le specifiche di progetto, sarà di tipo duale. Procediamo:

Vpp out = 2,82 x 4 = 11,28

Al valore ottenuto è bene aggiungere almeno 1 V, esso dovrebbe garantirci nei confronti della caduta di tensione sui dispositivi attivi posti nello stadio d'uscita dell'operazionale. Quindi:

11,28 + 1 = 12,5 (arrotondando)

Poiché il tutto è alimentato in modo duale possiamo subito considerare il valore di un singolo ramo d'alimentazione in valore assoluto (privo di segno):

12,5 / 2 = 6,25 VDC

Purtroppo il valore ricavato non è standard, senza troppi problemi possiamo scegliere un'alimentazione di +9 e -9 VDC. Veniamo ora alla scelta dell'operazionale. Esso sarà fornito dalle nostre scorte "casalinghe" e sarà un comune LM741C. Non ci resta che verificare se la resistenza di carico imposta dai dati di progetto (600 Ω) non richieda una corrente troppo elevata per il componente scelto. E già che ci siamo, tanto per essere sicuri, possiamo anche controllare se la tensione di alimentazione scelta è nei limiti previsti dal componente in oggetto. Consultando il datasheet, troviamo una corrente massima (definita di corto circuito) pari a 25 mA mentre la tensione d'alimentazione non deve eccedere i 36 VDC. E' bene precisare che quest'ultimo valore deve essere considerato come la somma dei valori dei singoli rami d'alimentazione. Questo significa che un operazionale con 36 VDC massimi d'alimentazione dovrà funzionare con un'alimentazione duale massima di +18 e -18 VDC (la metà). Tuttavia, la nostra applicazione rientra pienamente nei valori sopracitati.
Per la corrente sarà sufficiente un piccolo conto:

Irms-out = Vrms-in x A / 600
= 1 x 4 / 600 = 6,6 mArms

Il valore trovato è abbastanza lontano dai limiti massimi previsti per l'LM741C per cui… Tutto bene! La frequenza di taglio in-

TABELLA 3

FORMULE DI PROGETTO PER FILTRO NOTCH 12 DB X OTTAVA

fn	**= 1 / (C R 6,28)**
R	**= 1 / (6,28 C fn)**
C	**= 1 / (6,28 R fn)**

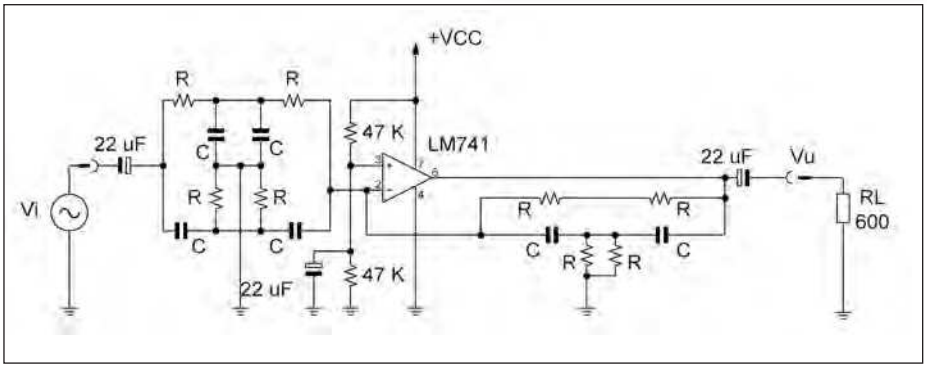


Figura 11: filtro notch di secondo ordine ad alimentazione singola.

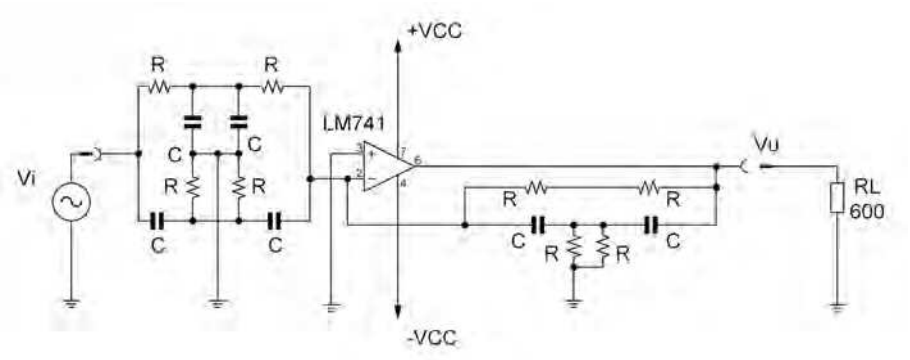


Figura 12: filtro notch di secondo ordine ad alimentazione duale.

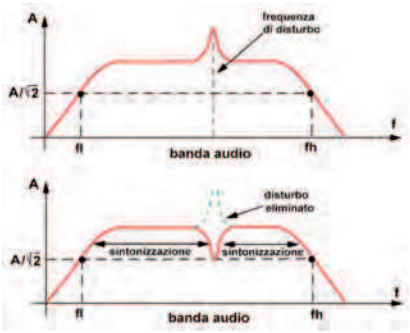


Figura 13: utilizzo di un filtro notch a frequenza variabile.

feriore, pari a 300 Hz, ci permetterà di ricavare i valori di R e C necessari alla sezione passa alto. Come già accennato, possiamo imporre il valore della capacità per poi ricavare il valore di R. Useremo la relazione relativa alla frequenza di taglio avente per incognita il valore resistivo:

$$R = 1 / (6{,}28 \times C \times ftl)$$

Nella relazione dovremmo inserire un valore di capacità arbitrario che, svolgendo la relazione, non dovrebbe permettere di ricavare valori resistivi né troppo grandi (superiori ad 1 M Ω) e né troppo bassi (inferiori a 100 Ω). Nel nostro caso proveremo con il valore standard di 47 nF:

$$R = 1 / (6{,}28 \times 0{,}000000047 \times 300) = 11{,}29\ K\Omega$$

Siamo stati abbastanza fortunati, il valore ricavato è molto prossimo al valore standard di 12 KΩ e, rinunciando ad un po' di precisione a vantaggio della semplicità, possiamo adottarlo nella nostra applicazione. Naturalmente, i valori di R e C ricavati sono validi per entrambe le sotto sezioni di primo e secondo ordine che, lo ribadiamo, devono avere la stessa frequenza di taglio. Procediamo con la stessa modalità per ricavare i valori di R e C usati dalla sezione passa basso la cui ft sarà, in questo caso, pari a 3,2 KHz. Visto che la frequenza di taglio superiore è circa 10 volte più grande di quella inferiore, la capacità con cui proveremo a calcolare la nostra R sarà esattamente 10 volte inferiore alla precedente: 4,7 nF. Impostiamo nuovamente la formula cambiando la frequenza di taglio:

$$R = 1 / (6{,}28 \times 0{,}0000000047 \times 3200) = 10{,}5\ K\Omega$$

Anche in questo caso è possibile arrotondare al valore standard di 10 K Ω. A questo punto siamo solo ad un passo per avere in mano tutti i valori necessari, coraggio! In effetti gli unici componenti ancora da determinare sono i resistori di controreazione il cui valore determina il guadagno in banda passante. Il valore di questi ultimi è pero fissato a priori dal tipo di risposta, nel nostro caso Butterworth, e dall'ordine del filtro. Osservando la tabella 5 della scorsa puntata possiamo subito guardare alla seconda riga ove è prevista un'attenuazione di 60 dB/decade ricavabile, come nel nostro caso, da un filtro di primo ordine seguito da un filtro di secondo ordine. I valori resistivi previsti per il filtro di secondo ordine sono, per entrambe le resistenze di controreazione, pari a 10 K Ω. E' ora possibile tracciare lo schema elettrico complessivo, esso è visibile in **figura 8**.

I FILTRI NOTCH (ELIMINA BANDA)

L'uso dei filtri notch avviene quando occorre eliminare da un segnale una ben determinata frequenza. Per questo motivo la banda passante di un filtro notch, salvo pochissime eccezioni, deve essere la più stretta possibile. Un uso classico per questo tipo di filtro è quello di filtrare i 50 Hz della rete dallo spettro di un segnale audio. Infatti, in impianti HI-FI costituiti da molti componenti e quindi da molte connessioni, è facile che il ronzio di rete possa introdursi nel percorso del segnale e, quindi, essere riprodotto dal sub-woofer. Anche se in teoria l'orecchio umano non dovrebbe percepire una frequenza così bassa, può accadere che il cono del sub-woofer cominci a vibrare a frequenza sub-sonica e questo fenomeno, avvertibile anche al tatto, disturba decisamente la riproduzione dei super bassi. Identiche applicazioni si hanno in settori dove la frequenza di rete costituirebbe un disturbo per il segnale da analizzare e/o misurare (elettromedicali). Qualche volta può essere utile eliminare la componente di prima armonica che, nel caso della rete europea, è di 100 Hz. Infatti, un segnale a 100 Hz è molto vicino al limite di udibilità e potrebbe disturbare di più del segnale fondamentale a 50 Hz. Si consideri, inoltre, che la tensione di rete raddrizzata produce, per l'appunto, un ripple che ha frequenza pari a 100Hz. Non è infrequente che piccoli residui di ripple possano "infilarsi" nei percorsi del segnale audio e sovrapporsi ai segnali udibili, con un filtro regolato a 100 Hz si possono evitare, in buona parte, questi effetti indesiderati. In figura 9 possiamo vedere la risposta in frequenza di un filtro notch a banda molto stretta. Risposte di questo tipo non sono però molto facili da ottenere. Per i motivi sopradetti eviteremo di presentarvi filtri notch di primo ordine e passeremo subito agli schemi applicativi relativi a filtri di secondo ordine. Vi sono molti schemi circuitali per ottenere reti filtranti elimina banda ma, come sempre, presenteremo il circuito che a parità di difficoltà offre migliori prestazioni. Un ottimo filtro notch, paragonabile ad una rete filtrante di secondo ordine, può essere ottenuto utilizzando due reti a ponte meglio definite a "doppio T" (vedi **figura 10**). Un circuito simile a quello di **figura 10**

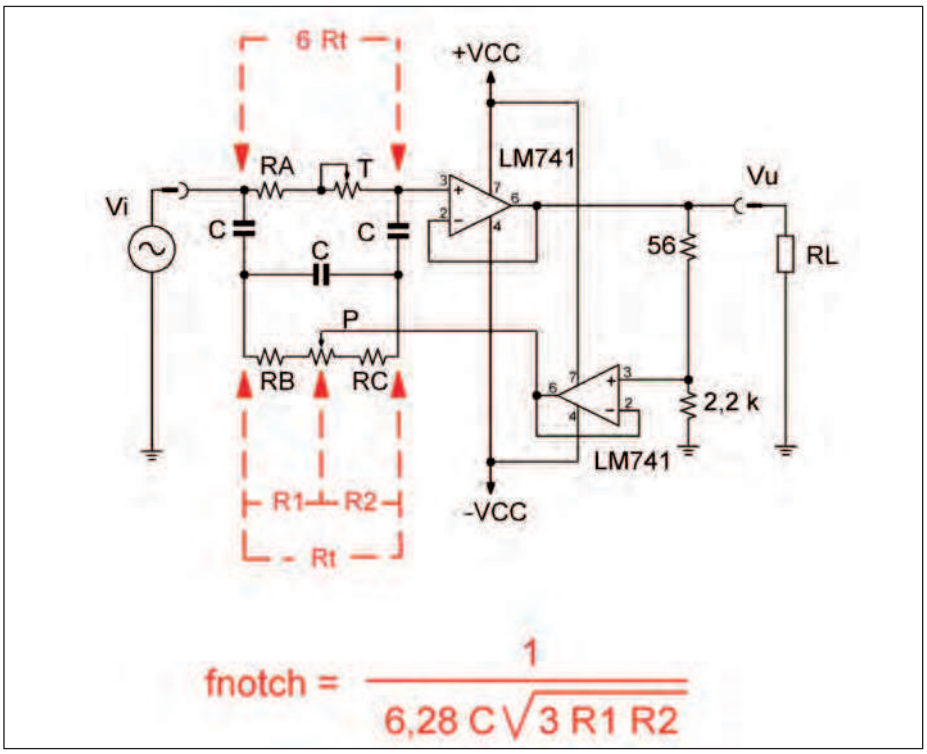


Figura 14: filtro notch a frequenza variabile di primo ordine (alimentazione duale).

può essere immaginato come una sorta di doppio filtro, passa basso e passa alto, avente stessa frequenza di taglio. In realtà, nella pratica circuitale, la rete "doppio T" di controreazione diretta (ovvero, quella che collega l'uscita al piedino invertente dell'operazionale) è leggermente modificata rispetto alla configurazione originale. Tuttavia la sostanza non cambia: è sempre una rete a "doppio T". In **figura 11** possiamo trovare un filtro notch di secondo ordine alimentato con tensione singola mentre, in **figura 12**, vi è lo stesso circuito per alimentazione duale. I filtri notch presentati hanno guadagno pari a 0 dB in banda passante. Si noterà che negli schemi presentati alcuni componenti sono posti in parallelo o in serie. E' naturale pensare che il circuito possa semplificarsi usando un singolo componente al posto di due. Tuttavia, l'efficacia filtrante di un filtro notch dipende in buona misura dall'uguaglianza dei valori R e C costituenti le celle a "doppio T".
Ne consegue che per evitare di rimanere "intrappolati" nei calcoli dei valori (e magari rischiare di non uscirne…) è meglio utilizzare, ove necessario, componenti in serie e parallelo garantendo ogni volta l'esatto dimezzamento o raddoppio del valore voluto. In **tabella 3** potete trovare le formule per il calcolo dei valori R e C. A volte potrebbe tornare utile avere un filtro notch che possa "sintonizzarsi" all'interno di un certo gruppo di frequenze e non su una sola. Questo potrebbe succedere, ad esempio, se noi vogliamo ascoltare una conversazione esterna tramite un microfono direzionale e contemporaneamente siamo disturbati da un rumore di fondo casuale.
Quest'ultimo potrebbe essere un motore a scoppio, il trillo di un uccellino e tanto altro ancora la cui frequenza, purtroppo, non potremmo stabilire a priori. In questi casi un filtro notch a sintonia variabile può tornare utile. Infatti, manovrando un potenziometro, potremmo eliminare, "ricercandolo" nella gamma di frequenze udibili, qualsiasi rumore dannoso per l'ascolto intaccando poco o niente la banda audio. La **figura 13** chiarisce il concetto esposto. In **figura 14** possiamo vedere lo schema applicativo di un filtro notch sintonizzabile con alimentazione duale. Esso è composto da una rete definita "a ponte" i cui valori resistivi devono rispecchiare fedelmente le equazioni di progetto (riportate nella stessa **figura 14**). Per questo motivo è stato previsto il trimmer T che dovrà tarare esattamente il valore del braccio resistivo superiore rendendolo assolutamente identico a sei volte il valore complessivo del braccio inferiore. Riguardo quest'ultimo si noti, osservando le annotazioni presenti nello schema elettrico, come nella relazione riguardante la frequenza di notch i valori delle resistenze dipendono dalla posizione del cursore di P. Naturalmente, l'operazione di taratura andrà fatta con alimentazione scollegata e mediante un multimetro digitale. Per P sarà necessario usare un potenziometro multigiri, di solito questi componenti vengono commercializzati con valori a potenza di 10 (1 K Ω, 10 K Ω e 100 K Ω) per cui è bene che tale condizione rientri nelle imposizioni di progetto. Non si pensi, tra l'altro, di ricoprire un range di frequenza elevatissimo con lo schema di **figura 14**. Anche se ciò in linea di massima sarebbe possibile in pratica, verso un estremo della corsa del potenziometro, il centraggio della frequenza diverrebbe molto critico in quanto sarebbe confinato in uno spazio di regolazione ridottissimo. Per coprire range di sintonia elevati è purtroppo necessario variare, a mezzo commutatore, i valori resistivi di RA, RB e RC in modo che il potenziometro, a metà corsa, si trovi su una frequenza circa doppia della precedente.

CONCLUSIONI

I lettori che hanno avuto la pazienza di seguirmi in queste tre puntate dovrebbero essere in grado di progettare filtri attivi per BF in modo autonomo e, mi auguro, per qualsiasi applicazione. Possono tuttavia, in special modo sul piano teorico, proseguire personali approfondimenti che li metteranno in condizioni di essere ancora più indipendenti e creativi nelle progettazioni di loro interesse. Se così fosse, vuol dire che queste pagine sono servite realmente a qualcosa. L'argomento, come ho accennato fin dalle prime battute, è vastissimo e complesso e richiede, così come per tante altre materie, impegno e motivazione. Buon lavoro! ❑

CODICE MIP 2804855

Corso di Python *(parte seconda)*

Valutazione delle ESPRESSIONI

Come in qualsiasi linguaggio di programmazione, anche in Python è possibile espressioni che l'interprete valuta in tempo reale e ne visualizza il risultato. Vediamo in dettaglio come funziona questo meccanismo

Una espressione è una combinazione di valori, variabili e operatori. Scrivendo un'espressione sulla riga di comando, l'interprete la valuta e mostra a video il risultato, come avviene nell'IDE NetBeans. La differenza tra "valutare un'espressione" e stamparne il valore, è sottile ma è di fondamentale importanza:

```
#Filename:var.py
messaggio = "come va? "
print messaggio
```

Output dopo F6:

come va?

Quando Python mostra il valore di un'espressione, usa lo stesso formato che si userebbe per inserirla: nel caso delle stringhe ciò significa che include le virgolette di delimitazione. L'istruzione print, invece, stampa il valore dell'espressione, che nel caso delle stringhe corrisponde al loro contenuto. Le virgolette sono, quindi, rimosse. In uno script un valore preso da solo è legale, anche se non fa niente e non produce alcun risultato.

GLI OPERATORI E OPERANDI

Gli operatori sono simboli speciali che rappresentano elaborazioni di tipo matematico, quali la somma e la moltiplicazione. I valori che l'operatore usa nei calcoli sono chiamati operandi. Le seguenti espressioni sono tutte legali in Python ed il loro significato dovrebbe esserti chiaro:

```
20+32 ore-1 ore*60+minuti minuti/60
5**2 (5+9)*(15-7)
```

L'uso dei simboli +, -, / e delle parentesi, sono uguali all'uso che se ne fa in matematica. L'asterisco (*) è il simbolo della moltiplicazione ed il doppio asterisco (**) quello dell'elevamento a potenza. Quando una variabile compare al posto di un operando, essa è rimpiazzata dal valore che rappresenta prima che l'operazione sia eseguita. Addizione, sottrazione, moltiplicazione ed elevamento a potenza fanno tutto ciò che potresti aspettarti, ma la divisione potrebbe non sembrare così intuitiva. L'operazione seguente ha infatti un risultato inatteso:

```
minuti=59
minuti/60
print minuti
```

Output dopo F6:

0

Il valore di minuti è 59, e 59 diviso 60 dà 0.98333, non zero. La ragione di questa differenza sta nel fatto che Python sta facendo una divisione tra numeri interi. Quando entrambi gli operandi sono numeri interi, il risultato è sempre un numero arrotondato all'intero inferiore (arrotondamento verso il basso), anche nel caso in cui il risultato sia molto vicino all'intero superiore. Una possibile soluzione a questo problema, potrebbe essere il calcolo della percentuale, piuttosto che del semplice valore decimale:

```
minuti*100/60
print minuti
```

Output dopo F6:
98

Ancora una volta il valore è arrotondato per difetto, ma almeno la risposta è approssimativamente corretta. Un'altra alternativa è l'uso della divisione in virgola mobile che tratteremo più avanti.

ORDINE DELLE OPERAZIONI

Quando più operatori compaiono in un'espressione, l'ordine di valutazione dipende dalle regole di precedenza. Python segue le stesse regole di precedenza usate in matematica:

• Parentesi: hanno il più alto livello di precedenza e possono essere usate per far valutare l'espressione in qualsiasi ordine. Dato che le espressioni tra parentesi sono valutate per prime, 2*(3-1) dà come risultato 4, e (1+1)**(5-2) dà 8. Puoi usare le parentesi per rendere più leggibile un'espressione come in (minuti*100)/60, anche se questo non influisce sul risultato.

• Elevamento a potenza: ha la priorità successiva così 2**1+1 fa 3 e non 4, e 3*1**3 fa 3 e non 27.

• Moltiplicazione e Divisione hanno la stessa priorità, superiore a somma e sottrazione. 2*3-1 dà 5 e non 4, e 2/3-1 fa -1, e non 1 (ricorda che la divisione intera 2/3 restituisce 0).

• Addizione e Sottrazione, anch'esse con la stessa priorità.

Gli operatori con la stessa priorità sono valutati da sinistra verso destra, così che nell'espressione minuti*100/60, la moltiplicazione è valutata per prima, ottenendo 5900/60, che a sua volta restituisce 98. Se le operazioni fossero state valutate da destra a sinistra il risultato sarebbe stato sbagliato: 59*1=59.

OPERAZIONI SULLE STRIGHE

In generale non puoi effettuare operazioni matematiche sulle stringhe, anche se il loro contenuto sembra essere un numero. Supponiamo che messaggio sia di tipo string, gli esempi proposti di seguito sono illegali:

messaggio-1 Ciao"/123 messaggio*"Ciao"
"15"+2

L'operatore + funziona con le stringhe anche se la sua funzione è diversa da quella cui siamo abituati in matematica: infatti, nel caso di stringhe, l'operatore + rappresenta il concatenamento, con l'aggiunta del secondo operando alla fine del primo. Per esempio:

```
frutta = banana
verdura = pomodoro
print frutta + verdura
```

Il risultato a video di questo programma è banana pomodoro. Lo spazio davanti alla parola pomodoro è parte della stringa ed è necessario per produrre lo spazio tra le due stringhe concatenate. Anche l'operatore * lavora sulle stringhe pur con un significato diverso rispetto a quello matematico: infatti, causa la ripetizione della stringa. Per fare un esempio, "Casa*3 è "CasaCasaCasa". Uno degli operandi deve essere una stringa, l'altro un numero intero. Da una parte questa interpretazione degli operatori + e * ha senso per analogia con l'addizione e la moltiplicazione in matematica. Così come 4*3 è equivalente a 4+4+4, ci aspettiamo che "Casa"*3 sia lo stesso di "Casa"+"Casa"+"Casa", ed effettivamente è così. D'altro canto c'è un particolare sostanziale che rende diverse la somma e la moltiplicazione di numeri e di stringhe.

CONSIDERAZIONI

Finora abbiamo guardato agli elementi di un programma (variabili, espressioni e istruzioni) prendendoli isolatamente, senza parlare di come combinarli. Una delle più utili caratteristiche dei linguaggi di programmazione è la loro capacità di prendere piccoli blocchi di costruzione e di comporli. Sappiamo già sommare e stampare dei numeri e possiamo fare le due operazioni nello stesso momento:

```
print 17+3
Output dopo F6:
20
```

In realtà l'addizione è stata portata a termine prima della stampa, cosicché le due operazioni non stanno avvenendo contemporaneamente. Qualsiasi operazione che ha a che fare con i numeri, le stringhe e le variabili può essere usata all'interno di un'istruzione di stampa. Hai già visto un esempio a riguardo:

```
print " Numero di minuti da mezzanotte:" , ore*60+minuti
```

Puoi anche inserire espressioni arbitrarie nella parte destra di un'istruzione di assegnazione:

```
percentuale = (minuti * 100) / 60
```

Questa capacità può non sembrare particolarmente impressionante, ma vedremo presto altri esempi in cui la composizione permette di esprimere elaborazioni complesse in modo chiaro e conciso. Attenzione: ci sono dei limiti su "dove" puoi usare certe espressioni. Per esempio, la parte sinistra di un'istruzione di assegnazione può solo essere una variabile e non un'espressione: minuti*60 = ore è illegale.

COMMENTI

Man mano che il programma cresce di dimensioni, diventa sempre più difficile da leggere. I linguaggi formali sono ricchi di significato e può risultare difficile capire a prima vista cosa fa un pezzo di codice o perché è stato scritto in un certo modo. Per questa ragione, è una buona idea aggiungere delle note ai tuoi programmi per spiegare con un linguaggio naturale cosa sta facendo il programma nelle sue varie parti. Queste note sono chiamate commenti e sono marcati dal simbolo #:

```
# calcola la percentuale di ore trascorse
percentuale =
(minuti*100)/60
```

In questo caso il commento appare come una linea a se stante. Puoi eventualmente inserire un commento alla fine di una riga:

```
percentuale = (minuti*100)/60 # attenzione: divisione intera
```

Qualsiasi cosa scritta dopo il simbolo # e fino alla fine della riga, viene trascurata nell'esecuzione del programma. Il commento serve al programmatore o ai futuri programmatori che dovranno usare questo codice. In questo ultimo esempio il commento ricorda al lettore che ci potrebbe essere un comportamento inatteso dovuto all'uso della divisione tra numeri interi.

FUNZIONI

Hai già visto un esempio di chiamata di funzione:

```
>>> type(32)
<type 'string'>
```

Il nome della funzione è type e mostra il tipo di valore della variabile. Il valore della variabile, che è chiamato argomento della funzione, deve essere racchiuso tra parentesi. E' comune dire che una funzione "prende" o "accetta" un argomento e "ritorna" o "restituisce" un risultato. Il risultato è detto valore di ritorno. Invece di stampare il valore di ritorno possiamo assegnarlo ad una variabile:

```
>>> betty = type(32)
>>> print betty
<type 'string'>
```

Come esempio ulteriore, la funzione id prende un valore o una variabile e ritorna un intero che agisce come un identificatore unico del valore:

```
>>> id(3)
134882108
>>> betty = 3
>>> id(betty)
134882108
```

Ogni valore ha un id unico che rappresenta dove è depositato nella memoria del computer. L'id di una variabile è l'id del valore della variabile cui essa si riferisce.

CONVERSIONE DI TIPO

Python fornisce una raccolta di funzioni interne che converte valori da un tipo all'altro. La funzione int prende ogni valore e lo converte, se possibile, in intero. Se la conversione non è possibile, mostra un messaggio d'errore:

```
>>> int(32)
32
>>> int(Hello)
ValueError: invalid literal for int(): Hello
```

int può anche convertire valori in virgola mobile in interi, ma ricorda che nel farlo tronca (cioè toglie) la parte decimale.

```
>>> int(3.99999)
3
>>> int(-2.3)
-2
```

La funzione float converte interi e stringhe in numeri in virgola mobile:

```
>>> float(32)
32.0
>>> float(3.14159)
3.14159
```

Infine, str converte al tipo stringa:

```
>>> str(32)
'32'
>>> str(3.14149)
'3.14149'
```

Può sembrare strano il fatto che Python distingua il valore intero 1 dal corrispondente valore in virgola mobile 1.0. Questo rappresentano effettivamente uno stesso numero ma appartengono a tipi differenti (rispettivamente intero e in virgola mobile) e, quindi, vengono rappresentati in modo diverso all'interno della memoria del computer.

FORZATURA DI TIPO

Per tornare ad un esempio del paragrafo precedente (la divisione di minuti per 60), ora che sappiamo convertire i tipi, abbiamo un modo ulteriore per gestire la divisione tra interi. Supponiamo di dover calcolare la frazione di ora che è trascorsa: l'espressione più ovvia, minuti/60, lavora con numeri interi, così il risultato è sempre 0 anche se sono trascorsi 59 minuti. Una delle soluzioni è quella di convertire minuti in virgola mobile e calcolare il risultato della divisione in virgola mobile:

```
>>> minuti = 59
>>> float(minuti) / 60.0
0.983333333333
```

In alternativa possiamo avvantaggiarci delle regole di conversione automatica dei tipi chiamate forzature di tipo. Nel caso di operatori matematici, se uno degli operandi è float, l'altro è automaticamente convertito a float:

```
>>> minuti = 59
>>> minuti / 60.0
0.983333333333
```

Convertendo il denominatore a valore in virgola mobile forziamo Python a calcolare il risultato di una divisione in virgola mobile. ❑

CODICE MIP 2804847

fare elettronica

Vieni a trovarci su Facebook! siamo già più di 1500!!!

aggiornamenti e novità in tempo reale

articoli gratuiti per i fan

anteprime e anticipazioni

eventi

contatti con gli autori

inviaci i tuoi consigli e i tuoi commenti

condividi i contenuti con i tuoi amici

DIVENTA PROTAGONISTA DELLA TUA RIVISTA PREFERITA!

www.facebook.com/farelettronica

CODICE MIP 2808441

di GIOVANNI DI MARIA

La comunicazione *in* RS485

Questo articolo ha lo scopo di studiare come avviene la comunicazione RS485 tra dispositivi a microcontrollore PIC, utilizzando il mikroBasic PRO

RS485 è una comunicazione *multipoint* che permette a più dispositivi di essere collegati ad un unico bus. Il mikroBasic PRO per PIC fornisce un insieme di routine di libreria per lavorare comodamente con il sistema RS485, utilizzando un'architettura Master/Slave. I dispositivi Master e Slave possono scambiarsi pacchetti di informazioni. Ognuno di questi pacchetti contiene byte di sincronizzazione, byte CRC, byte d'indirizzo e, naturalmente, dati. Ogni Slave ha un indirizzo unico e riceve solo pacchetti inviati ad esso. Lo Slave non può mai iniziare una comunicazione.

DISTANZA TRA MASTER E SLAVE

Come detto prima, la comunicazione RS485 permette il colloquio tra una unità Master e una o più unità Slave. Essa avviene semplicemente tramite due fili. Per brevi distanze, è possibile usare qualsiasi tipologia di connessione, anche un normale doppino telefonico.
Ma se la lunghezza raggiunge molte centinaia di metri, occorre utilizzare cavi appositi. Teoricamente si possono far comunicare tra loro dispositivi distanti anche 1200 metri (1,2 Km!).
Personalmente ho realizzato prototipi che colloquiavano tranquillamente a 400 metri, senza alcun problema, usando cablaggi di fortuna.

SCHEMA ELETTRICO DI PRINCIPIO

Le funzioni RS485 richiedono la presenza fisica di un modulo UART. I pins della UART necessitano anche di una interfaccia transceiver RS485, come l'integrato LTC485, MAX 485, MAX 487 o si-

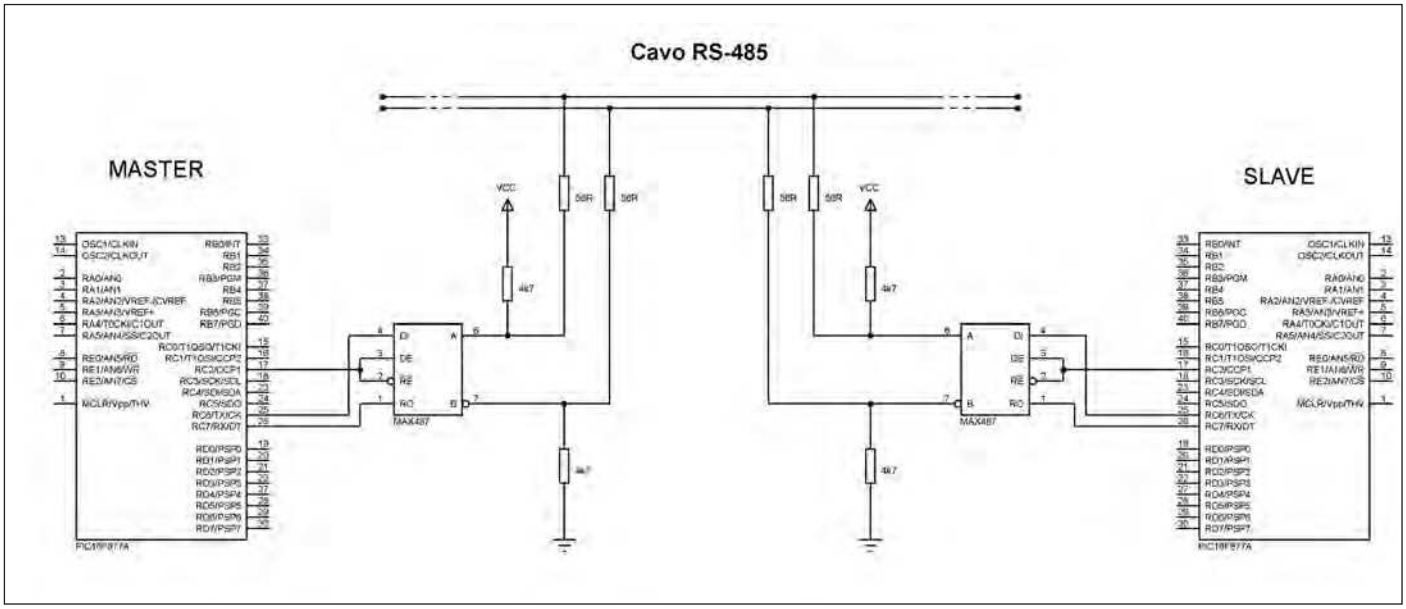


Figura 1: schema di principio di connessione tra Master e Slave.

Figura 2: il transceiver MAX 485

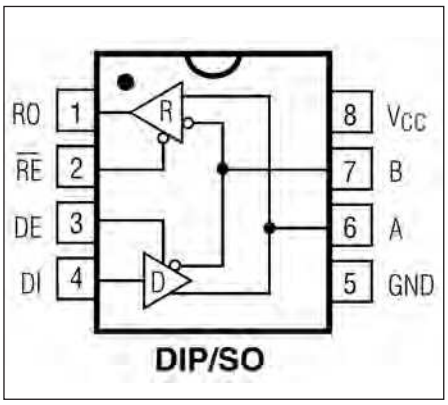


Figura 3: pinout del MAX 485

Pin MAX481/MAX483/ MAX485/MAX487/ MAX1487 DIP/SO	Pin MAX481/MAX483/ MAX485/MAX487/ MAX1487 µMAX	NAME	FUNCTION
1	3	RO	Receiver Output: If A > B by 200mV, RO will be high; If A < B by 200mV, RO will be low.
2	4	$\overline{RE}$	Receiver Output Enable. RO is enabled when $\overline{RE}$ is low; RO is high impedance when $\overline{RE}$ is high.
3	5	DE	Driver Output Enable. The driver outputs, Y and Z, are enabled by bringing DE high. They are high impedance when DE is low. If the driver outputs are enabled, the parts function as line drivers. While they are high impedance, they function as line receivers if $\overline{RE}$ is low.
4	6	DI	Driver Input. A low on DI forces output Y low and output Z high. Similarly, a high on DI forces output Y high and output Z low.
5	7	GND	Ground
—	—	Y	Noninverting Driver Output
—	—	Z	Inverting Driver Output
6	8	A	Noninverting Receiver Input and Noninverting Driver Output
—	—	A	Noninverting Receiver Input
7	1	B	Inverting Receiver Input and Inverting Driver Output
—	—	B	Inverting Receiver Input
8	2	Vcc	Positive Supply: 4.75V ≤ Vcc ≤ 5.25V
—	—	N.C.	No Connect—not internally connected

Figura 4: funzione dei pin del MAX 485.

mili. La comunicazione avviene, almeno per il PIC 16F877, attraverso le porte RC6 (TX) e RC7 (RX). Esaminiamo lo schema elettrico di principio in **figura 1**. I due microcontrollori sono connessi tra loro attraverso due fili.
L'interfaccia MAX 487, o simili, potenzia il segnale e fornisce il supporto RS485. Grazie ad essa, possono essere raggiunte distanze notevoli.
La porta RC7 (RX) del PIC è connessa al pin 1 (RO) del transceiver, mentre la porta RC6 (TX) è connessa al pin 4 (DI) del transceiver. Infine, la porta RC2 del PIC è connessa ai piedini 2 (-RE) e 3 (DE) del transceiver.
Completano la logica del collegamento le resistenze di pull-up e pull-down da 4,7k e, soprattutto, le due resistenze di linea a basso valore, da 56 ohm.
Questi collegamenti sono uguali sia per il Master che per lo Slave. Completano la connessione alla linea i pin 6 (A) e 7 (B) del transceiver, collegati agli altri dispositivi per lo scambio dei pacchetti.

IL TRANSCEIVER MAX 485 E ALTRI MODELLI

Il MAX481, MAX483, MAX485, MAX487–MAX491 e MAX1487 sono transceiver a bassa potenza per la comunicazione RS485 e RS422. Ogni modello contiene un driver e un ricevitore. Hanno la capacità di ridurre al minimo la EMI e ridurre le riflessioni causate da cavi interrotti, garantendo trasmissioni di dati prive di errori fino a 250kbps. Alcuni modelli consentono di trasmettere dati fino a 2.5Mbps.

ELENCO FUNZIONI E PROCEDURE MIKROBASIC

Quello che segue è un elenco delle funzioni e procedure implementate in mikroBasic, atte a gestire in toto la comunicazione RS485:

- RS485Master_Init
- RS485Master_Receive
- RS485Master_Send
- RS485Slave_Init
- RS485Slave_Receive
- RS485Slave_Send

RS485Master_Init()

Inizializza il microcontrollore e imposta il suo funzionamento come Master.

RS485Master_Receive(dim byref data_buffer as byte[20])

Riceve i messaggi dallo Slave. I messaggi ricevuti sono composti da diversi bytes. Il programmatore deve separare manualmente i dati ed elaborare i risultati. Il buffer ricevuto è composto da 7 bytes, alla versione attuale del compilatore.
Ognuno di essi ha una funzione ben precisa:

- data[0..2]: contenuto del messaggio
- data[3]: numero dei bytes del messaggio ricevuto (1–3)
- data[4]: è impostato a 255 quando il messaggio è ricevuto
- data[5]: è impostato a 255 se avviene un errore di comunicazione
- data[6]: indirizzo dello Slave che ha spedito il messaggio

RS485Master_Send(dim byref data_buffer as byte[20], dim datalen as byte, dim slave_address as byte)
Invia i messaggi allo Slave. I parametri della procedura sono i seguenti:
- data_buffer: messaggio da spedire
- datalen: Numero di bytes trasmessi che compongono il messaggio (1-3)
- slave_address: indirizzo dello Slave

RS485Slave_Init(dim slave_address as byte)
Inizializza il microcontrollore e imposta il suo funzionamento come Slave. Il suo parametro informa semplicemente il numero di indirizzo assegnato allo Slave.

RS485Slave_Receive(dim byref data_buffer as byte[20])
Riceve un messaggio dal Master. Se l'indirizzo dello Slave non corrisponde all'indirizzo inviato dal Master, il messaggio non sarà considerato. Il buffer ricevuto è composto, alla versione attuale del compilatore, da 6 bytes. Ognuno di essi ha una funzione ben precisa:
- data[0..2]: contenuto del messaggio
- data[3]: numero dei bytes del messaggio ricevuto (1–3)
- data[4]: è impostato a 255 quando il messaggio è ricevuto
- data[5]: è impostato a 255 se avviene un errore di comunicazione

RS485Slave_Send(dim byref data_buffer as byte[20], dim datalen as byte)
Spedisce un messaggio al Master. I parametri della procedura sono i seguenti:
- data_buffer: messaggio da spedire
- datalen: numero di bytes per la trasmissione. (0-3)

ESPERIENZA: 3 PULSANTI E 3 LED

Il nostro studio sulla comunicazione RS485 è concentrato sul colloquio unilaterale tra Master e Slave. Due unità sono collegate tra loro. La prima unità (Master) è dotata di tre pulsanti. La seconda unità (Slave) è dotata di tre diodi Led colorati diversamente.
La pressione del primo pulsante determina l'accensione del primo Led, la pressione del secondo pulsante determina l'accensione del secondo Led e la pressione del terzo pulsante determina l'accensione del terzo Led. Le accensioni sono mu-

LISTATO 1

```
program master

rem ——-Dichiarazione variabili———-
dim dat as byte[20]
dim RS485_rxtx_pin as sbit at RC2_bit
dim RS485_rxtx_pin_direction as sbit at TRISC2_bit


main:

rem ————Porte di I/O————-
trisb=7

rem ———Azzera PORTE I/O———-
porta=0
portb=0
portc=0
portd=0

rem ———Inizializza RS-485 come MASTER———-
UART1_Init(9600)    ' initialize UART module
Delay_ms(100)
RS485Master_Init()  ' intialize MCU as

while true

  rem  Se si preme tasto n. 1 Manda codice 1-2-3 allo slave
   if portb.0=1 then
      dat[0]=1
      dat[1]=2
      dat[2]=3
      RS485Master_Send(dat,3,0)
   end if

  rem  Se si preme tasto n. 2 Manda codice 4-5-6 allo slave
   if portb.1=1 then
      dat[0]=4
      dat[1]=5
      dat[2]=6
      RS485Master_Send(dat,3,0)
   end if

  rem  Se si preme tasto n. 3 Manda codice 7-8-9 allo slave
   if portb.2=1 then
      dat[0]=7
      dat[1]=8
      dat[2]=9
      RS485Master_Send(dat,3,0)
   end if

   delay_ms(10)
wend

end.
```

un CONTROLLORE *INdUSTRiALE*

PROGRAMMABILE IN C

Modular Industrial Controller

MOACON

Guarda
il video:

MASSIMA *flessibilità* per la TUA applicazione!

- DESIGN MODULARE
- PROGRAMMABILE IN C
- BASATO SU PROCESSORE ARM A 32 BIT
- DOWNLOAD E DEBUG DEL FIRMWARE VIA USB
- AMBIENTE DI SVILUPPO GRATUITO

CODICE MIP 2808445

MODULI DISPONIBILI

digital I/O, RELAY OUTPUT, convertitori Analog-to-Digital e Digital-to-Analog, **CONTROLLO MOTORE**, *Monitoraggio Temperatura*, COMUNICAZIONE RS-232, Ethernet

Distribuito da **elettroshop.com** brilliant electronics since 1998 *info@elettroshop.com – tel. 0266464794 - www.elettroshop.com*

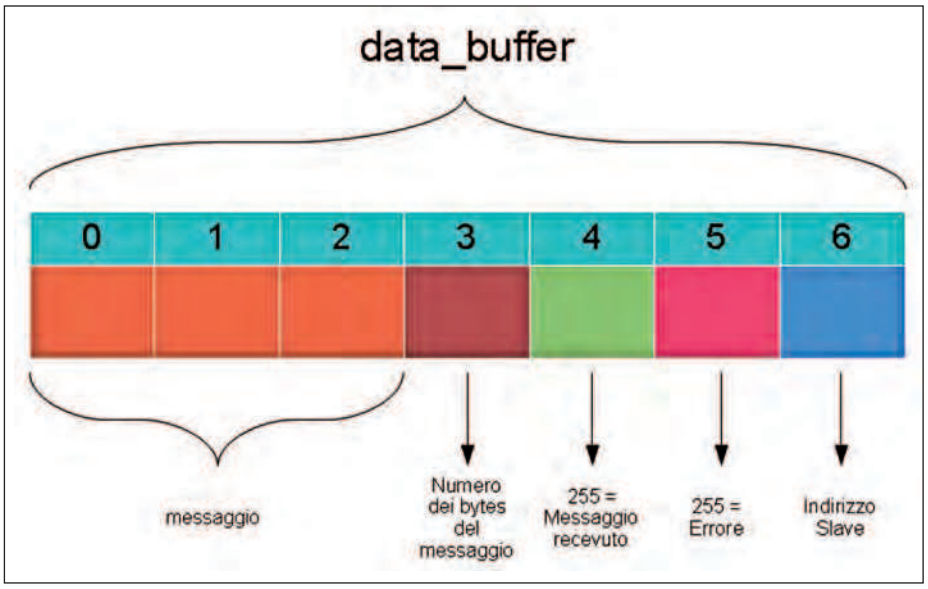


Figura 5: struttura del pacchetto data_buffer.

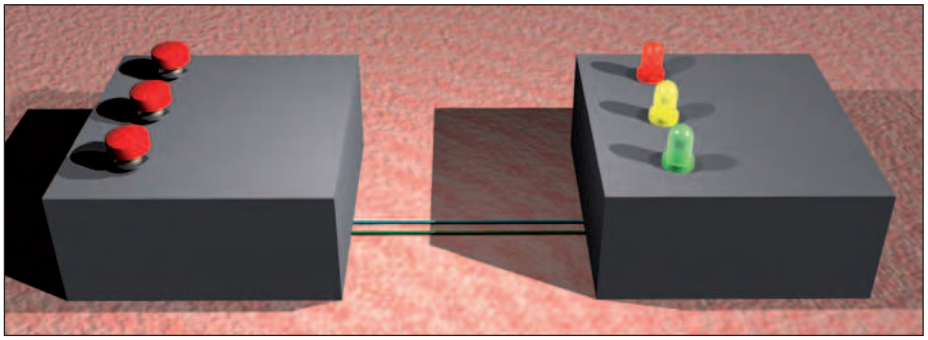


Figura 6: il prototipo 3D relativo all'esperimento.

tuamente esclusive. Lo stato iniziale prevede che tutti i Led siano spenti. Le due unità sono collegate tra loro semplicemente con due fili. Supponiamo, infine, che il codice dello Slave sia 0.

Schema elettrico

Lo schema elettrico è molto semplice. Il Master ha a corredo 3 pulsante n/a che, se premuti, portano ad ALTO lo stato logico delle porte di ingresso RB0, RB1, RB2. Se la pressione non è effettuata, lo stato logico è BASSO, grazie alle resistenze di pull-down da 10k.
Lo Slave, a sua volta, ha 3 diodi Led collegati rispettivamente alle porte di uscita RB0, RB1 e RB2, attraverso le relative resistenze di limitazione da 330 ohm.

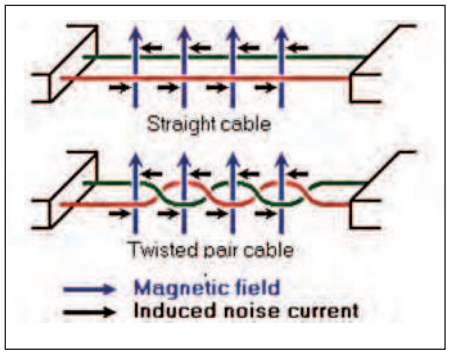


Figura 7: differenza di prestazioni tra cavo dritto e cavo incrociato.

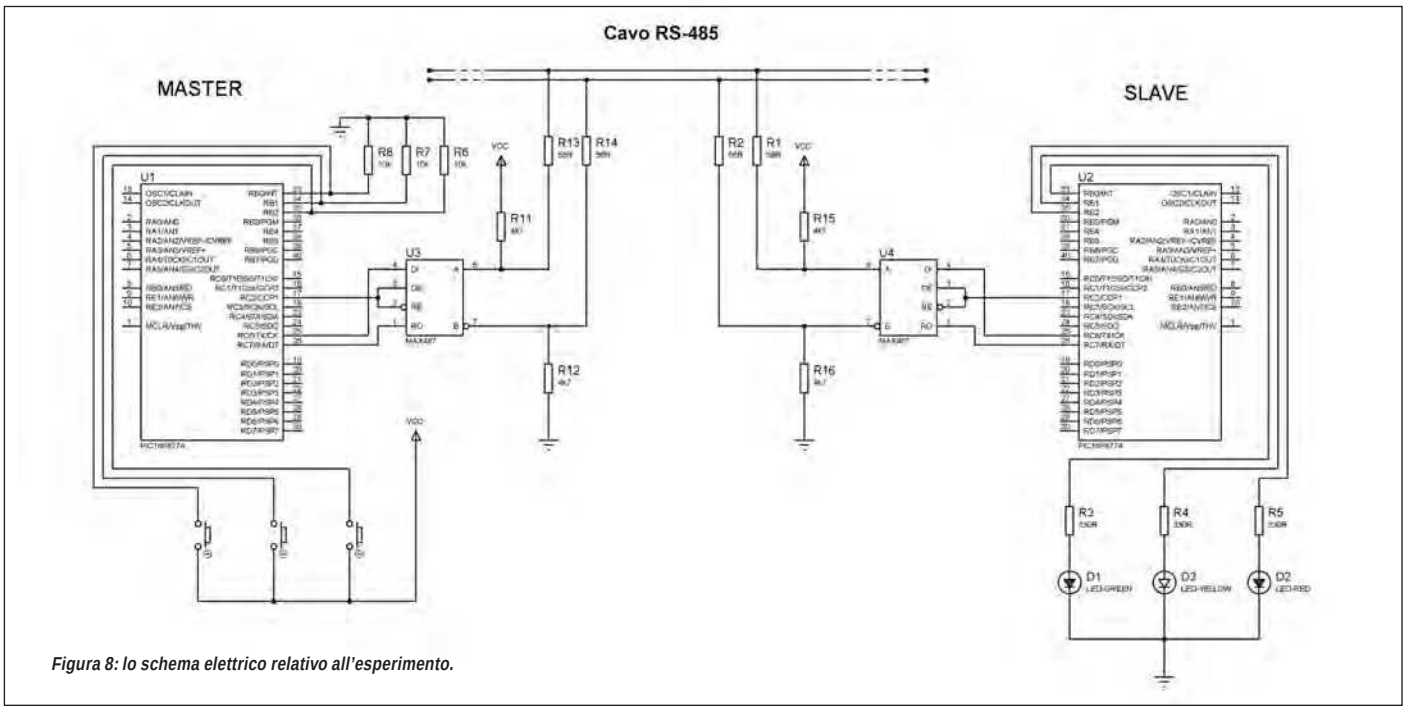


Figura 8: lo schema elettrico relativo all'esperimento.

Il programma Master

Il programma per la gestione del master è riportato nel **listato 1**. Dopo la definizione delle variabili, delle porte e dopo l'azzeramento delle stesse, viene effettuato il settaggio della comunicazione RS485. Un ciclo infinito scandisce continuamente, in polling, lo stato dei tre pulsanti. Se il pulsante 1 viene premuto, il codice 1-2-3 è inviato allo Slave.
Se il pulsante 2 viene premuto, il codice 4-5-6 è inviato allo Slave. Se il pulsante 3 viene premuto, il codice 7-8-9 è inviato allo Slave. Questo tipo di codifica assicura una comunicazione perfetta, esente da errori e priva di incertezze, anche se il cavo è estremamente lungo.

Il programma Slave

Anche il programma per la gestione dello Slave è semplice ed è riportato nel **listato 2**. Dopo le solite dichiarazioni e inizializzazioni della porte e delle variabili, un ciclo infinito contiene il controllo dei messaggi ricevuti dal Master. Lo Slave ha codice 0. Il primo led si illumina se, e solo se, il messaggio è stato ricevuto correttamente (dat[4]=255), se non vi sono stati errori di comunicazione (dat[5]<>255) e se il codice ricevuto è 1-2-3 (dat[0]=1 and dat[1]=2 and dat[2]=3). Lo stessa filosofia di funzionamento è applicata agli altri due diodi Led.

Il sistema di codifica

Per assicurare una comunicazione sicura, esente da errori, anche a grandi distanze, è consigliabile impostare un messaggio a 3 bytes. In questo modo si inviano i codici 1-2-3 (tasto 1) oppure 4-5-6 (tasto 2) oppure ancora 7-8-9 (tasto 3) nei byte corrispondenti a dat[0], dat[1] e dat[2]. Lo Slave a sua volta riceverà tali codici e imposterà il comportamento del programma di conseguenza

Cavi RS485

Per aumentare l'immunità al rumore, e quindi incrementare le distanze di comunicazione, è opportuno utilizzare cavi speciali. Con due fili normali, le differenze di potenziale del segnale possono facilmente variare ed il rumore è facilmente raccolto dal cavo. In un cavo dritto, il rumore scorre nella stessa direzione del segnale. Se il cavo è incrociato, in alcune zone della linea il rumore è opposto alla corrente. Per tale motivo, il rumore totale è molto minore rispetto a quello captato dal cavo dritto.

Conclusioni

Come si è visto, implementare un sistema di comunicazione RS485 è abbastanza semplice. Volendo, e se le esigenze lo richiedono, si possono utilizzare le procedure opposte e le corrette funzioni per inviare i messaggi dallo Slave. ❑

CODICE MIP 2804879

LISTATO **2**

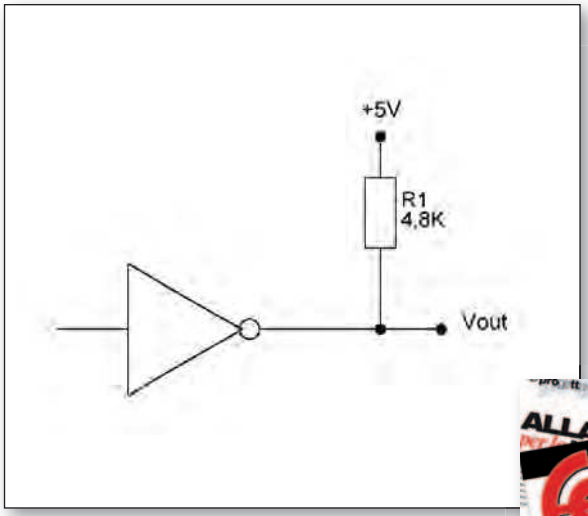
```
program slave

'———Dichiarazione variabili———-
dim dat as byte[20]
dim rs485_rxtx_pin as sbit at RC2_bit
    rs485_rxtx_pin_direction as sbit at TRISC2_bit

main:

'——Porte di I/O——
trisb = 0   'Uscite
portb = 0

'——-Inizializza RS 485 per SLAVE——-
UART1_Init(9600)      ' initialize UART1 module
RS485Slave_Init(0)   ' Initialize MCU as slave 0
delay_ms(10)


while TRUE
   '—-Azzera Flag di errore——
   dat[4] = 0    ' ensure that message received flag is 0
   dat[5] = 0    ' ensure that error flag is 0

   '——Riceve Messaggio da slave———
   RS485Slave_Receive(dat)

   '—-Se riceve codice 1-2-3 accende Led 1————-
   if (dat[4]=255) and (dat[5]<>255) and (dat[0]=1) and
(dat[1]=2) and (dat[2]=3) then
      portb = 1
   end if

   '—-Se riceve codice 4-5-6 accende Led 2————-
   if (dat[4]=255) and (dat[5]<>255) and (dat[0]=4) and
(dat[1]=5) and (dat[2]=6) then
      portb = 2
   end if

   '—-Se riceve codice 7-8-9 accende Led 3————-
   if (dat[4]=255) and (dat[5]<>255) and (dat[0]=7) and
(dat[1]=8) and (dat[2]=9) then
      portb = 4
   end if


   delay_ms(1)
wend

end.
```

elettro quiz 312

rispondi & vinci

Le risposte ai quiz "Base" e "Avanzato" vanno inviate esclusivamente compilando il modulo su www.farelettronica.com/eq specificando la parola chiave ***"inductor"****.*

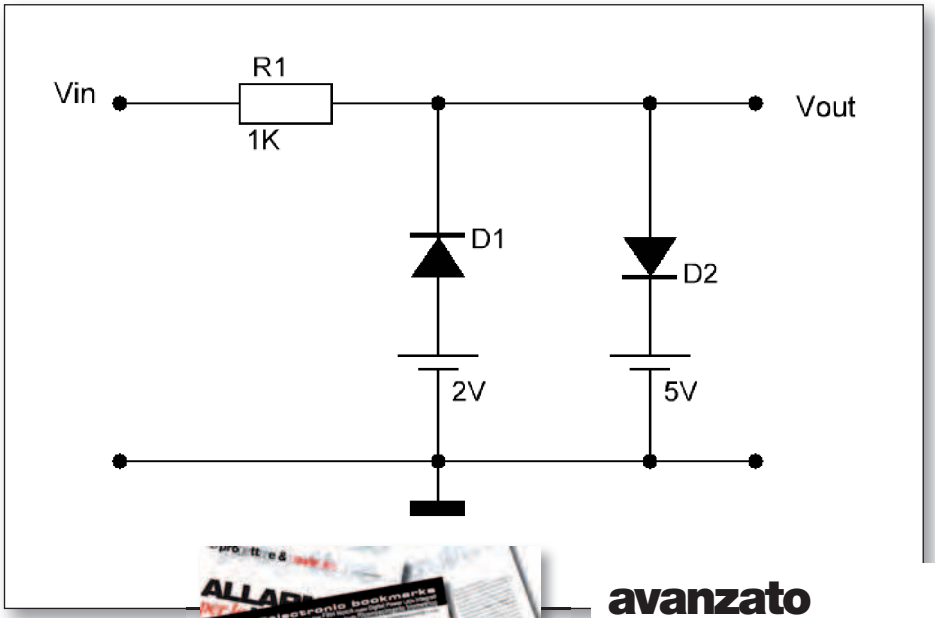
Le risposte ed i vincitori (previa autorizzazione) sono pubblicati alla pagina www.farelettronica.com/eq a partire dal 15 del mese successivo alla pubblicazione sulla rivista. A tutti i partecipanti verrà assegnato un buono sconto del 10% (validità 3 mesi dalla data di assegnazione) utilizzabile per un prossimo acquisto su www.ieshop.it



base

La porta NOT nel circuito di figura è di tipo open-collector. Sapendo che la tensione di uscita a livello basso è 0,2V quanto vale la corrente nella resistenza di pull-up in questo stato?

Se rispondi correttamente potrai vincere l'abbonamento (o rinnovo) al Club di Fare Elettronica per 1 anno.



avanzato

Dato il circuito non lineare di figura, determinare il valore della tensione di uscita quando Vin=12V, Vin=3V e Vin=-5V.

Per i più bravi in palio l'abbonamento annuale a Fare Elettronica.

convertitori USB-seriale

compatti, performanti e convenienti

MULTI-1U
Convertitore USB 1.1 a **1 Porta** RS232 o 422/485

MULTI-2U
Convertitore USB 1.1 a **2 Porte** RS232 o 422/485

MULTI-8U
Convertitore USB 1.1 a **8 Porte** RS232 o 422/485

MULTI-4U
Convertitore USB 1.1 a **4 Porte** RS232 o 422/485

CODICE MIP 2808449

SystemBase

Per maggiori informazioni visita il sito **www.elettroshop.com** oppure chiama il numero **02/66504794**

di DAVIDE BOSCOLO

In quest'articolo analizzeremo il funzionamento del Trigger di Schmitt: un circuito per la realizzazione di soglie di tensione e impiegato in tutti i circuiti che richiedano di lavorare con valori variabili in ingresso

IL TRIGGER di Schmitt

Il trigger di Schmitt è uno dei circuiti elettronici più utilizzati in ambito elettronico. Termostati, gestione di livello, stadi di ingresso di porte logiche, sono solo alcuni degli esempi in cui probabilmente si trova un dispositivo di questo tipo. La facilità nel reperire il materiale per la costruzione, il basso costo e l'elevata affidabilità, ne fanno un sistema in grado di gestire efficacemente il variare aleatorio del segnale da controllare. "Misurato" il livello d'ingresso e confrontato il valore con la soglia impostata, il trigger è in grado di attivare uscite digitali, avviare pompe o effettuare qualsiasi manovra determinata precedentemente.

PERCHÉ SERVE UN CIRCUITO A DOPPIA SOGLIA?

Lavorando con soglie, con livelli di tensione che sorpassati nel fronte di salita o discesa devono attivare o disattivare un certo dispositivo, è molto complesso. É facile capire che con questo sistema, seppure efficace, si trova qualche difficoltà nell'impiego: una leggera modifica alla variabile misurata, per un istante brevissimo, provoca la commutazione del circuito da uno stato all'altro. Prendiamo in esame un sistema di riscaldamento con termostato: quando la temperatura scende sotto un preimpostato livello, soglia, il circuito avvierà la procedura di riscaldamento, ma non appena superato il valore di temperatura massimo, soglia, il sistema dovrà procedere a spegnere la ventilazione calda. Facile e intuitivo come impianto, se non fosse che utilizzando una valore di temperatura fisso, ci si troverà al punto che il sistema si accenderà e spegnerà ripetutamente per brevi istanti, ciò perché: raggiunto un punto di rendimento elevato, in cui tutte le parti sono in temperatura, non appena si accenderà la ventilazione l'ambiente si riscalderà immediatamente raggiungendo il livello e spegnendo di conseguenza l'impianto. Ma dopo poco tempo, con la dispersione termica, la temperatura scenderà sotto il consentito, riavviando il sistema e procedendo con un nuovo ciclo, vanificando il risparmio energetico ipotizzato dall'inserimento del termostato. Per perfezionare il sistema, e riuscire ad avere, quindi, un funzionamento ottimale, è ideale l'inserimento di un differenziale. Il trigger di Schmitt rende disponibile utilizzare due differenti soglie: una per l'accensione e una per lo spegnimento. Ritornando al caso in esame, un volta impostata una temperatura minima, si procederà con l'inserimento del differenziale, avviando il riscaldamento quando viene raggiunto il valore minimo di temperatura, spegnendolo al valore massimo.

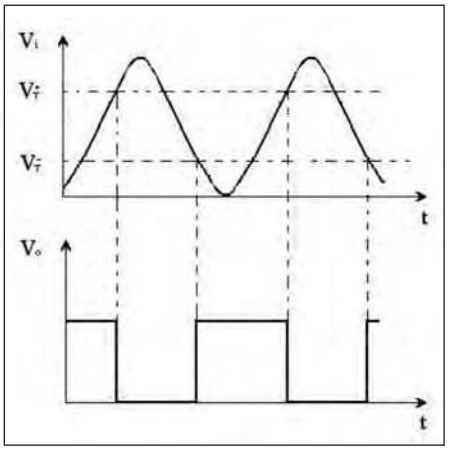


Figura 1: come risulta chiaramente dal diagramma temporale, il trigger di Schmitt può essere utilizzato come squadratore, ossia per trasformare in onda quadra un segnale con andamento qualsiasi.

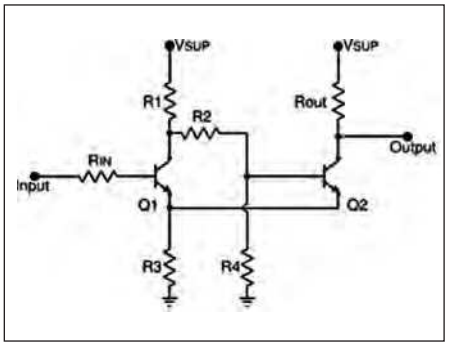


Figura 2: schema elettrico di un Trigger di Schmitt a BJT.

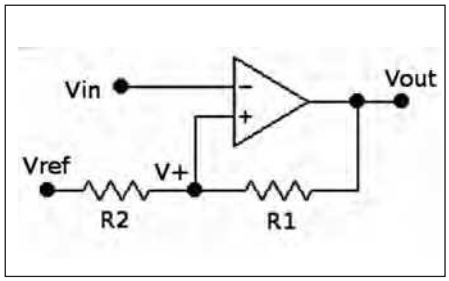


Figura 3: schema elettrico ad operazionale di un comparatore con isteresi invertente.

CAMPI D'IMPIEGO

Non esiste un vero e proprio campo d'impiego per questo circuito, può essere adattato per il funzionamento in qualsiasi situazione, in quanto uno dei circuiti principi dell'elettronica. La peculiarità di disporre di due soglie differenti, una per l'eccitazione e l'altra per la diseccitazione, lasciano libero arbitrio al progettista per qualsiasi tipo di sistema abbia la necessità di realizzare. Per esempio, nella caratteristica di ingresso di una comune porta logica, la zona di transizione fra il livello alto e quello basso è piuttosto incerta e non molto brusca. Il trigger di Schmitt presenta al contrario due soglie di

commutazione molto nette. Partendo da una tensione d'ingresso V_{in} nulla, l'uscita V_{ut} del trigger rimane alta finché V_{in} non supera la cosiddetta soglia superiore VT^+. Dal livello basso l'uscita ritorna, invece, alta solo quando la tensione di ingresso scende al di sotto di una soglia inferiore VT^-, diversa e più bassa dell'altra.

CIRCUITO A BJT

Lo schema più semplice, ma anche meno efficace per realizzare un circuito a doppia soglia, è mediante l'impiego di due transistor in cascata. La maglia costituita da R1 e R2 è un partitore di tensione che fornisce la tensione di base al transistore Q1. Peraltro, l'NPN Q1 ha effetto sul partitore che fornisce una tensione più bassa quando questi è chiuso. Cosicché, la tensione di soglia di commutazione di stato dell'uscita dipende dallo stato della corrente. In condizioni di segnale basso in ingresso, Q1 è interdetto e Q2 lavora. La tensione di base del transistor Q2 è stabilita dal partitore di tensione citato. Q1 può lavorare solo se il segnale d'ingresso si eleva un pò oltre la caduta di tensione ai capi di R3. Quando Q1 lavora, Q2 è interdetto, in quanto la tensione fornita è troppo bassa per far lavorare il transistor. In condizioni di segnale d'ingresso più alto, Q1 conduce mentre Q2 è interdetto. Quando il segnale d'ingresso si riduce, Q1 inizia ad interdirsi aumentando la tensione di base di Q2. Raggiunto un certo livello, Q2 aumenta la corrente riducendo ulteriormente il potenziale base-emettitore di Q1 e interdicendolo. Nello stato "1" la tensione in uscita è circa uguale a V_{sup}, ma nello stato logico "0" questa delle volte risulta ancora troppo elevata per essere elettricamente uno zero. Per raggirare ciò, normalmente, si possono utilizzare due soluzioni: o si inseriscono amplificatori supplementari in cascata all'uscita o si cambiano le tensioni di riferimento di alimentazione, operando quindi con sorgenti anche negative.

IL CIRCUITO A OPERAZIONALI

Per l'implementazione, se non si ha la necessità di utilizzare un dispositivo a logica programmabile, il circuito che reagisce con maggior efficacia viene con-

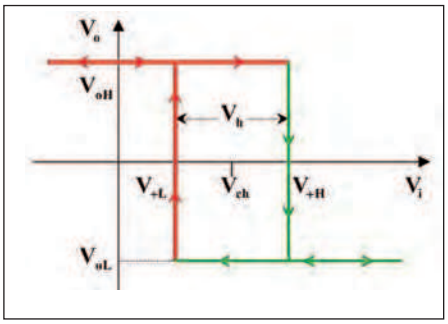


Figura 4:grafico d'isteresi di un Trigger di Schmitt ad operazionale invertente.

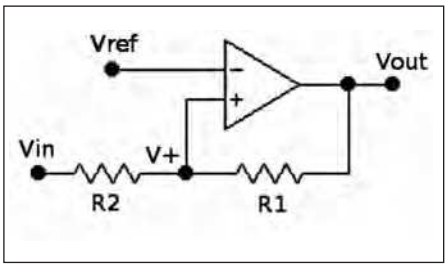


Figura 5: schema elettrico ad operazionale di un comparatore con isteresi non invertente.

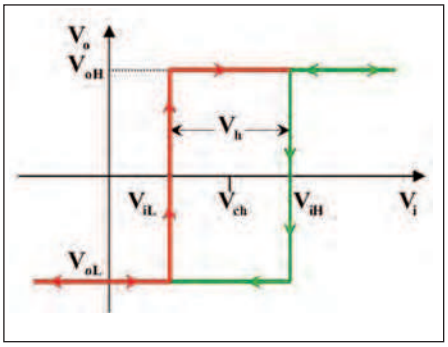


Figura 6:grafico d'isteresi di un Trigger di Schmitt ad operazionale non invertente.

cepito con un operazionale. L'operazionale, vista la peculiarità di fare la differenza tra i due segnali nei morsetti d'ingresso, con qualche resistenza e una leggera retroazione, è efficacissimo per lavorare e svolgere la funzione di comparatore con isteresi. La retroazione positiva viene realizzata applicando all'ingresso non invertente una parte della tensione di uscita, tramite un partitore resistivo. L'uscita commuta ogni qualvolta la tensione $V_- =$ Vin sull'ingresso invertente, uguaglia la tensione V_+ sull'ingresso non invertente. La tensione V_+ dipende dalla tensione d'uscita che può assumere due valori, V_{oH} e V_{oL}, pertanto, si avranno due tensioni di soglia, Vsup e Vinf .

$$Vsup = \frac{R2}{R1+R2} V_{OH}$$

$$Vinf = \frac{R2}{R1+R2} V_{OL}$$

Quando V_{in}, crescendo, uguaglia e tende a superare Vsup, prevale l'ingresso invertente su quello non invertente, provocando la commutazione dell'uscita da V_{oH} a V_{oL}, e la tensione sull'ingresso non invertente passa da Vsup a Vinf, ossia la tensione di soglia subisce una drastica variazione. Una nuova commutazione dell'uscita si avrà quando la tensione d'ingresso Vin, decrescendo, uguaglia e tende a scendere al di sotto di Vinf, prevale l'ingresso non invertente su quello invertente provocando la commutazione dell'uscita da V_{oL} a V_{oH}, e la tensione sull'ingresso non invertente passa da Vinf a Vsup, ossia la tensione di soglia subisce un'altra drastica variazione. Il circuito, come evidenziato dalla caratteristica d'uscita, presenta una isteresi di ampiezza V_h, ossia le commutazioni dell'uscita avvengono per valori della tensione di soglia diversi tra loro ed esiste un intervallo delle tensioni d'ingresso per le quali l'uscita rimane allo stato precedente. Si può concludere che l'impiego della retroazione positiva spinge Vo a commutare più velocemente; inoltre, se le due soglie di commutazione sono sufficientemente distanziate e maggiori dei picchi della tensione di rumore, vengono eliminate le false commutazioni. La differenza $V_h = V_{sup} - V_{inf}$ fra le due soglie di commutazione prende il nome di ampiezza dell'isteresi. Per evitare false commutazioni occorre, dunque, che V_h sia maggiore dell'ampiezza picco-picco dei disturbi; così l'ampiezza dell'isteresi indicherà l'ampiezza del segnale di rumore che può essere tollerata o il margine di rumore. Il centro dell'isteresi in questo caso è zero.

Circuito per trigger di Schmitt invertente

Se si vuole un centro dell'isteresi diverso da zero, si collega la resistenza R2, non a massa, ma ad una opportuna tensione di riferimento come in figura. La commu-

PER *approfondire...*

- http://www.scuolaelettrica.it/elettrotecnica/differe2.php
- http://www.massimomarini.net/DOWNLOAD/Amplificatore%20operazionale%20da%20Wiki.pdf
- http://en.wikipedia.org/wiki/Schmitt_trigger

tazione dell'uscita si ha ogni qualvolta la tensione V_+ uguaglia la tensione $V_- = V_{ref}$. Alla tensione V_+ contribuiscono due cause, V_{out} e V_i; applicando il principio di sovrapposizione degli effetti, si ha:

$$V_{+H} = \frac{R2}{R1+R2} \cdot V_{oH} + \frac{R2}{R1+R2} \cdot V_{ref}$$

$$V_{+L} = \frac{R2}{R1+R2} \cdot V_{oL} + \frac{R2}{R1+R2} \cdot V_{ref}$$

Il centro dell'isteresi, supponendo $V_{oH} = -V_{oL}$, è:

$$V_{ch} = \frac{V_{+H} + V_{+L}}{2} = \frac{R1}{R1+R2} \cdot V_{ref}$$

L'ampiezza dell'isteresi è:

$$V_{ch} = V_{+H} - V_{+L} = \frac{R2}{R1+R2} \cdot 2V_{oH}$$

Circuito per trigger di Schmitt non invertente

Il comparatore con isteresi non invertente si ottiene invertendo tra loro le tensioni d'ingresso e di riferimento, come in figura. La commutazione dell'uscita si ha ogni qualvolta la tensione V_+ uguaglia la tensione $V_- = V_{ref}$. Alla tensione V_+ contribuiscono due cause, V_{out} e V_i; applicando il principio di sovrapposizione degli effetti, si ha:

$$V+ = \frac{R1}{R1+R2} \cdot V_i + \frac{R2}{R1+R2} \cdot V_{out}$$

Supponendo la tensione d'uscita a livello alto, $V_{out} = V_{oH}$, la tensione V_+ deve essere necessariamente maggiore della tensione $V_- = V_{ref}$.

$$V_{out} = V_{oH}$$
$$V+ = \frac{R1}{R1+R2} \cdot V_i + \frac{R2}{R1+R2} \cdot V_{out} > V_{ref}$$

La commutazione dell'uscita da V_{oH} a V_{oL} si avrà quando, diminuendo la tensione d'ingresso V_i la tensione V_+ uguaglia e tende a scendere al di sotto della tensione V_{ref}, per un valore della tensione d'ingresso V_{iL}, che è anche il valore della tensione di soglia inferiore. Si impone, quindi, che:

$$V+ = \frac{R1}{R1+R2} \cdot V_{iL} + \frac{R2}{R1+R2} \cdot V_{oH} = V_{ref}$$
$$V_{iL} = \frac{-R2}{R1} \cdot V_{oH} + \frac{R1+R2}{R1} \cdot V_{ref}$$

Supponendo la tensione d'uscita a livello basso, $V_{out} = V_{oL}$, la tensione V_+ deve essere necessariamente minore della tensione $V_- = V_{ref}$. Se

$$V_{out} = V_{oL}$$
$$V_+ = \frac{R1}{R1+R2} \cdot Vin + \frac{R2}{R1+R2} \cdot V_{oL} < Vref$$

la commutazione dell'uscita da V_{oL} a V_{oH} si avrà quando, aumentando la tensione d'ingresso V_i la tensione V_+ uguaglia e tende a superare la tensione V_{ref}, per un valore della tensione d'ingresso V_{iH}, che è anche il valore della tensione di soglia superiore. Si impone, quindi, che:

$$V+ = \frac{R1}{R1+R2} \cdot V_{iH} + \frac{R2}{R1+R2} \cdot V_{oL} = V_{ref}$$
$$V_{iH} = \frac{-R2}{R1} \cdot V_{oL} + \frac{R1+R2}{R1} \cdot V_{ref}$$

Il centro dell'isteresi, supponendo $V_{oH} = -V_{oL}$, è:

$$V_{ch} = \frac{V_{iH} + V_{iL}}{2} = \left(1 + \frac{R2}{R1}\right) V_{ref}$$

l'ampiezza dell'isteresi invece ha valore:

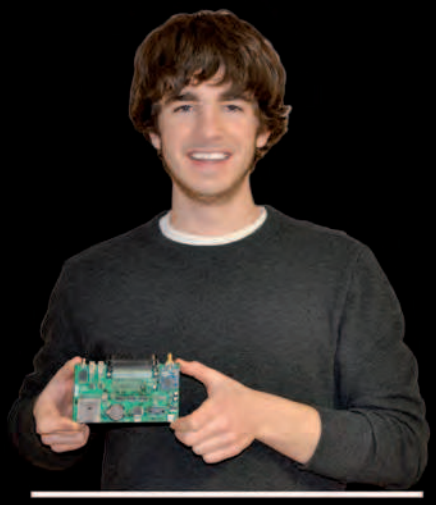
$$V_h = V_{iH} - V_{iL} = \frac{R2}{R1} \cdot 2V_{oH}$$

CONCLUSIONI

Recentemente, con il mio amico Moreno, mi sono dedicato alla modifica di un alimentatore da banco stabilizzato autocostruito. L'elettronica interna prevedeva un trasformatore a doppio primario, da commutare al passaggio di una soglia, facendo dissipare in maniera più contenuta lo stabilizzatore interno. Inizialmente si pensava di utilizzare un transistor con un partitore di tensione collegato alla base, ma vista l'entità del carico in uscita ci si è ritrovati nella stessa situazione del termostato descritto ad inizio articolo, con in più la necessità di realizzare un circuito a soglia con isteresi. Chiunque impiega del tempo nello studio dell'elettronica, o chi semplicemente ci lavora, ha ormai capito a sue spese che l'ambito teorico non va a diretto contatto con la pratica, e ciò che su carta può sembrare facile e scontato, spesso in pratica si tramuta in un progetto da rivedere. Il Trigger di Schmitt non è una soluzione, ma un circuito da adottare a priori quando si lavora con segnali da processare variabili nel tempo. ❑

CODICE MIP 2804863

Vincitore: Thumper by Harrison Pham - *Radio player via internet con registrazione MP3 e funzioni di playback. Funzionalità inesistenti in nessun prodotto commerciale e costruito a un costo molto più basso!*

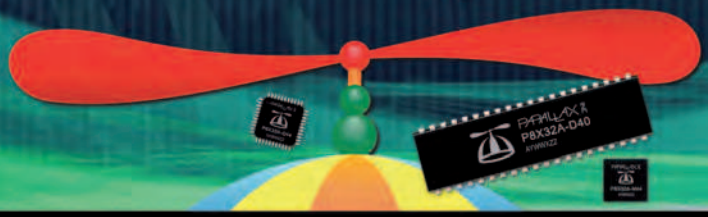


Secondo posto: DAQPac by Ryan David - *Data logger automotive per gli appassionati di motorsport, con tutte le funzionalità necessarie ad un pilota per migliorare sia la guida che le performance del veicolo.*

Terzo posto: Sphinx by Michael Park - *Sphinx by Michael Park - Sphinx è un compilatore Spin basato su Propeller che compila programmi complessi (inclusi quelli contenenti codice Propeller ASM) come la Parallax TV e gli oggetti grafici. Sphinx esegue anche funzioni tipiche di un sistema operativo. Fornisce una shell a linea di comando, un editor di testi, utilità per i dischi, e un sistema di I/O residenti su memoria (cog).*

Quale è la tua applicazione con Propeller?

Con ben otto processori a 32 bit (cogs) in un unico chip e il controllo deterministico dell'intero sistema, il chip Propeller multicore genera ispirazione. Ecco i vincitori del recente contest sui progetti basati sul Propeller.



Menzione d'onore: PropFloppy by Sal Mustafa - *un controller floppy con un'interfaccia terminal e la possibilità di memorizzare immagini su una memory stick.*

MO: Prop RTOS by Peter Van der Zee - *Un kernel multithread di facile implementazione in un assembler cog, con la possibilità di schedulare facilmente le temporizzazioni dei trigger per ciascun thread.*

MO: Rock Band Robot basata su Propeller by Adam Stienecker - Un sistema che può automaticamente suonare una Rock Band sulla xBox 360 utilizzando la video analysis con un chip Propeller che controlla la piattaforma.

Distribuito da elettroshop.com
tel. 02 66504794 - fax 02 66508225
www.elettroshop.com/propeller

Friendly microcontrollers, legendary resources.™

PARALLAX INC
www.parallax.com

CODICE MIP 2808457

Parallax ed il logo Parallax sono marchi registrati di Parallax Inc.

"ParallaxInc" on Twitter, Facebook, and YouTube

di MARIANO SEVERI

Corso SystemVerilog *(parte seconda)*

Procedure *task* E FUNZIONI

I blocchi procedurali sono insiemi di istruzioni eseguite in seguito all'attivazione del blocco stesso. Eccom come crearli e richiamarli

Nella precedente puntata abbiamo introdotto alcuni concetti basilari del SystemVerilog, ovvero i moduli (visti come contenitori statici di funzioni, task ed altri oggetti per la descrizioni dei blocchi logici) e i diversi tipi di dati supportati (con la distinzione, specifica del linguaggio, tra le variabili e le net e le relative modalità di assegnazione di tipo blocking e nonblocking). In questa puntata vedremo, invece, come descrivere i blocchi procedurali che si trovano all'interno dei moduli. Tali blocchi procedurali sono insiemi di istruzioni eseguite in seguito all'attivazione del blocco stesso. Le modalità di attivazione dei blocchi differiscono tra loro in funzione del tipi di questi. Sono previsti i seguenti tipi:

- initial
- final
- always, always_comb, always_latch, always_ff
- task
- function

Per i primi tre tipi di procedure può talvolta essere adottata la denominazione di processo. Un modulo può instanziare diversi processi che saranno eseguiti concorrentemente dal simulatore (secondo lo schema descritto nella precedente puntata del corso), per modellizzare il parallelismo di esecuzione delle architetture hardware. I processi possono contenere a loro volta chiamate a funzioni e task.

L'INSIEME DI ISTRUZIONI DEL SYSTEMVERILOG

La **tabella 1** riporta l'insieme di tipi di istruzioni supportate dal SystemVerilog, unitamente ad alcuni semplici esempi per ognuna di esse. Per brevità, eviteremo di seguito di discuterle nel dettaglio; si rimanda per questo ad uno qualunque dei test indicati nei riferimenti. La maggior parte di tali istruzioni è ereditata dal Verilog che, a sua volta, segue fortemente la sintassi del linguaggio C.

Più istruzioni possono essere raggruppate insieme mediante le parole chiavi *begin…end* e *fork…join*. Nel primo caso, le istruzioni sono eseguite sequenzialmente mentre nel secondo concorrentemente. Il listato 1 riporta un semplice esempio, tratto dal testo di Spear [2], che evidenzia la differenza tra i due costrutti. Tale codice produrrà il seguente output:

@0: start fork…join example
@10: sequential after #10
@10: parallel start
@20: parallel after #10
@40: sequential after #30
@50: sequential after #10
@60: parallel after #50
@60: after join
@140: final after #80

Il simulatore, infatti, inizia ad eseguire sequenzialmente le istruzioni, inviando

LISTATO **1**

```
1. initial begin
2.   $display("@%0d: start fork...join example", $time);
3.   #10 $display("@%0d: sequential after #10", $time);
4.   fork
5.     $display("@%0d: parallel start", $time);
6.     #50 $display("@%0d: parallel after #50", $time);
7.     #10 $display("@%0d: parallel after #10", $time);
8.     begin
9.         #30 $display("@%0d: sequential after #30",
$time);
10.        #10 $display("@%0d: sequential after #10",
$time);
11.   end
12.join
13.$display("@%0d: after join", $time);
14.#8
```

TABELLA 1: TIPI DI ISTRUZIONI SUPPORTATE DAL SYSTEMVERILOG

TIPI DI ISTRUZIONI IN SYSTEMVERILOG	ESEMPIO
blocking_assignment	r = #1 a; // assegna a ad r dopo 1 time unit
nonblocking_assignment	r <= #1 a; // assegna a ad r dopo 1 time unit
procedural_continuous_assignment	assign r = #1 a; // assegna a ad r dopo 1 time unit
case_statement	case(a) 0,1: $display(“0 or 1”); 2: $display(“2”); 4: $display(“4”); endcase
conditional_statement	if ((a==0) ll (a==1)) $display(“0 or 1”); else if (a == 2) $display(“2”); else if (a == 4) $display(“4”);
inc_or_dec_expression	i++; // incrementa I di 1
subroutine_call_statement	void’(my_func(42)); // chiama my_func con 42 come argument. Esegue il cast a void del valore ritornato per ignorarlo
disable_statement	disable u1.always1.t1 // disabilita il task t1 del blocco procedurale always1 del modulo u1
event_trigger	-> e1; // trigger l’evento e1

loop_statement	for_loop : for (int i = 15; i >= 0; i—) // for loop
	$display(i);
	while_loop : do $display(i); while (i—) // do…while loop

jump_statement	return; // ritorno da chiamata a funzione
	break;
	continue;

procedural_timing_control_statement	## 5; // attende 5 cicli (clocking events) del clock di default
wait_statement	wait (posedge clk); // attende il fronte positivo del segnale di clock
procedural_assertion_statement	assert (myfunc(a,b)) count1 = count + 1; else ->event1; // se myfunc ritorna 0 incrementa count, altrimenti triggera event1

randsequence_statement	// crea una sequenza di transazioni
	randsequence (stream)
	stream : cfg_read := 1 l io_read := 2 l mem_read := 5;
	cfg_read : { cfg_read_task; } l { cfg_read_task; } cfg_read;
	mem_read : { mem_read_task; } l { mem_read_task; } mem_read;
	io_read : { io_read_task; } l{ io_read_task; } io_read;
	endsequence

randcase_statement	// scelta pesata
	randcase
	1: len = $urandom_range(0, 2); // 10%: 0, 1, or 2
	8: len = $urandom_range(3, 5); // 80%: 3, 4, or 5
	1: len = $urandom_range(6, 7); // 10%: 6 or 7
	endcase

expect_property_statement	expect(@(posedge clk) a ##1 b ##1 c) else $error(“expect failed”); // attende fino a che non è verificata la sequenza specificata (a ##1 b ##1 c)

su output i messaggi di testo indicati alle righe 2 e 3. Quindi esegue il blocco *fork...join* compreso tra le righe 4 e 12. In tale blocco, le istruzioni sono schedulate nel tempo mediante ritardo #. Essendo le istruzioni del blocco eseguite parallelamente, i messaggi schedulati prima nel tempo saranno inviati prima su output; così il messaggio associato alla riga 7 è visualizzato prima di quello alla riga 6. D'altra parte, il blocco *fork...join* include a sua volta un insieme di istruzioni *begin..end* (righe 8-11) che sono eseguite tra loro sequenzialmente; in questo caso, quindi, l'istruzione alla riga 10 viene eseguita solo dopo che è stata eseguita quella alla riga 9, con il risultato che il relativo messaggio viene inviato su output solo dopo 50 unità di tempo.

PROCEDURE INITIAL E ALWAYS

Le procedure di tipo *initial* ed *always* sono eseguite all'istante zero di simulazione. Le prime sono eseguite una sola volta mentre le seconde riprendono dall'inizio l'esecuzione una volta raggiunta l'ultima istruzione. Il **listato 2** riporta un esempio di tali procedure. Sono mostrati, in particolare, due diversi modi di controllare l'esecuzione delle procedure, utilizzando istruzioni di attesa e controllo su evento. Nel caso delle procedure *always*, l'insieme di eventi che ne attiva l'esecuzione è definita *sensitivity list*.

Nel primo caso (righe 1-8), si utilizzo un controllo su evento che arresta l'esecuzione della sequenza *always* al suo termine, mentre nel secondo la sequenza è arrestata all'istruzione desiderata mediante istruzione *wait*. Nel primo caso, si descrive un contatore con ingresso di reset asincrono attivo alto e segnale di abilitazione sincrono; il contatore si incrementa sul fronte positivo del segnale di clock. Nel secondo, invece, è descritta l'uscita di un contatore che rimane inattiva fino a che il segnale di reset non viene rimosso; il contatore si incrementa per i successivi tre cicli di clock ed infine si arresta. Quindi il processo attende che il segnale di reset sia asserito nuovamente per ripetere la sequenza.

LISTATO **2**

```
1.   always @ (posedge arst, posedge clk)
2.     begin
3.       if (arst)
4.         q = '0;
5.       else
6.         if (en)
7.           q = q + 1;
8.     end
9.
10.  always
11.    begin
12.      q = '0;
13.      wait (arst==0);
14.      repeat(3)
15.         begin
16.            q = q + 1;
17.            wait (posedge clk);
18.         end
19.      wait (arst==1);
20.    end
```

LISTATO **3**

```
1. always_comb a = b & c;
2.
3. always_comb d <= #1ns b & c;
4.
5. always_latch
6.   if(l) q <= d;
7.
8. always_ff @(posedge clock iff reset == 0 or posedge re-
set) begin
9. r1 <= reset ? 0 : r2 + 1;
10.end
```

PROCEDURE ALWAYS_COMB, ALWAYS_FF ED

Le procedure di tipo *always_comb* sono utilizzate per descrivere logica combinatoria. Sono eseguite automaticamente all'istante zero, dopo l'inizializzazione dei blocchi *initial* ed *always*, per consentire la consistenza dei valori di assegnazione con quelli degli ingressi. La *sensitivity list* è definita implicitamente sulla base dell'insieme di variabili ed espressioni di selezione che siano lette all'interno della procedura stessa o di qualsiasi funzione ivi richiamata (ad eccezione, evidentemente, delle variabili dichiarate localmente alle funzioni o di espressioni scritte dalla procedura stessa o dalle funzioni richiamate). Variabili assegnate all'interno di procedure *always_comb* non possono tuttavia essere assegnate in altre istruzioni. Il **listato 3** riporta un esempio di procedura *always_comb* per la descrizione di due porte di tipo AND2; si noti come la prima istruzione è un'assegnazione i tipo *blocking* mentre la seconda di tipo *nonblocking* e schedulata dopo 1 ns.

Le procedure di tipo *always_latch* sono utilizzate per la descrizione di latch; modalità di attivazione e limitazioni sono del tutto identiche a quelle delle procedure *always_comb*. Il **listato 3** riporta un esempio di tale tipo di processo. Per descrivere, infine, strutture logiche sequenziali sin-

tetizzabili possono essere utilizzate le procedure always_ff. Le restrizioni da considerare, in questo caso, sono che non possono essere utilizzate istruzioni blocking (il che, per inciso, è ragionevole intendendo descrivere costrutti sintetizzabili) e che è possibile definire un solo evento di controllo. Un esempio di procedura always_ff utilizzata per la descrizione di un contatore con ingresso di reset asincrono è inclusa nel **listato 3**.

TASK E FUNZIONI

Funzioni e task sono blocchi procedurali che consentono di eseguire insiemi comuni di istruzioni in diversi punti del codice mediante semplice chiamata. Si differenziano nei seguenti aspetti:

- una funzione deve essere eseguita in un tempo di simulazione uno, ovvero non può contenere istruzioni con controllo di tempo (come, ad esempio, assegnazioni *blocking* o attese su eventi)
- una funzione non può richiamare un task mentre un task può contenere chiamate ad altri task e funzioni.

Funzioni e task possono essere di tipo statico od automatico; ove non specificato, si assume il tipo statico. Funzioni e task statici condividono lo stesso spazio di memoria per tutte le chiamate ad essi presenti in una istanza di un modulo. Funzioni e task automatici allocano, invece, uno spazio di memoria unico. Il SystemVerilog consente, tuttavia, la definizione di variabili automatiche in funzioni o task statici o oggetti statici in funzioni o task automatici.

Il **listato 4** riporta un esempio di dichiarazione di task e funzioni. Nel caso di funzioni, è previsto il tipo *void* per l'oggetto ritornato. I parametri formali delle procedure possono essere di tipo:

- *in*, la variabile locale viene inizializzata con il valore dell'argomento al momento della chiamata della procedura
- *out*, il valore della variabile locale viene assegnato all'argomento della chiamata all'uscita dalla procedura
- *inout*, la variabile viene inizializzata con il valore dell'argomento al momento della chiamata della procedura ed il suo valore viene ri-assegnato all'argomento della chiamata all'uscita dalla stessa
- *ref,* l'argomento viene passato mediante *reference*, ovvero chiamante e procedura condividono la stessa rappresentazione dell'argomento.

Nel caso di passaggio di argomenti mediante *reference*, non è consentito il casting del tipo.

Tale passaggio mediante *reference* è, in effetti, piuttosto simile all'utilizzo dei puntatori come argomenti di funzioni in linguaggio C. Viene tipicamente adottato nel caso di procedure che abbiano vettori come argomento (nel caso s'intenda evitare che la procedura chiamata modifichi l'array, è possibile definire l'argomento *const*) od in presenza di diversi thread che accedano concorrentemente alla stessa variabile.

Un esempio di quest'ultimo caso si trova nel **listato 5** (vedi [2]). Poiché l'argomento *data* è passato mediante *ref*, il task *bus_read* può modificare la variabile *data* locale al modulo durante la sua stessa esecuzione. Quindi, l'istruzione alla riga 16 riconosce l'evento sulla linea *data* non appena il suo valore cambia. Passando, invece, l'argomento per valore al task *bus_read*, la variabile locale *data* sarebbe aggiornata soltanto all'uscita dal task e, quindi, l'istruzione 17 non visualizzerebbe all'istante corretto il nuovo valore della stessa. Allo stesso modo, qualora il task *bus_read* modificasse più volte la sua variabile *data*, soltanto l'ultimo valore verrebbe visualizzato su output.

CONCLUSIONI

In questa puntata abbiamo discusso i costrutti, ovvero blocchi procedurali, funzioni e task, che il SystemVerilog rende disponibili per la descrizione delle funzionalità logiche e la manipolazione dei dati. Tali concetti, sebbene con sintassi diverse, sono in realtà comuni agli altri linguaggi di descrizione hardware, VHDL e Verilog. Come anticipato nell'introduzione al nostro corso, la caratteristica distintiva del SystemVerilog risiede, piuttosto, nel supporto di una programmazione ad oggetti mediante i concetti di classi ed ereditarietà. Questo consente la descrizione di strutture logiche ad un livello più alto favorendo la configurabilità ed il riutilizzo dei modelli. Nella prossima puntata vedremo appunto come descrivere e utilizzare in SystemVerilog le classi. ❑

CODICE MIP 2804843

PER *approfondire...*

[1] 1800-2009 IEEE Standard for SystemVerilog
Unified Hardware Design, Specification, and Verification Language

[2] Spear,C "SystemVerilog for Verification - A Guide to Learning the Testbench Language Features" – Springer, 2006

[3] Sutherland,S. Davidmann, S. & Flake, Peter "SystemVerilog For Design" – Springer, 2006

[4] Bergeron,J "WritingTestbenches using System Verilog" – Springer, 2003

[5] il listato 5 lo trovi su www.farelettronica.com

LISTATO 4

```
1.  task mytask1 (output logic [31:0] x, input logic y);
2.  ...
3.  endtask : mytask1
4.
5.  function integer myfunct1 (output logic [31:0] x,
    input logic y);
6.  …
7.  endfuntion : myfunct1
```

l'angolo di Mr A.KEER

I registers FILE

Continua in questa puntata la descrizione dei componenti correlati con l'elemento di memoria di tipo D; la struttura del Registro, fondamentale nel progetto digitale, viene ora organizzata in batteria, al fine di disporre di dispositivi che possano essere scritti e letti contemporaneamente

L'importanza funzionale degli elementi di memoria di tipo D è decisamente degna di nota: la loro presenza sta alla base di numerosi dispositivi, a partire da quelli chiamati *Registri* (che contengono da 4 fino a 64 D-Type), trattati in dettaglio la volta scorsa; in sostanza ciascuno di essi può essere inteso come una memoria sulla quale può essere letta o scritta una sola parola binaria, di solito formata da tanti bit quanti sono i Flip-Flops coinvolti.Abbiamo sottolineato come questa struttura sia fondamentale nell'ambito dei microcontrollori, non solo come porta d'uscita (per assicurare in modo stabile l'informazione alle periferiche specializzate a ricevere dati, come una batteria di Led o di Relè), ma anche come memoria temporanea interna agli stessi processori (al fine di facilitare l'elaborazione aritmetica o logica dell'informazione assunta o lasciata sul Bus Dati). In questa puntata ci occuperemo di memorie più grandi, realizzate raggruppando più *Registri* e organizzandoli in modo da poter essere letti e/o scritti a piacere; le strutture così ottenute sono piuttosto numerose e sono catalogate in due categorie: i *Registers File* e le *Random Access Memories* (sinteticamente note come *RAMs*).

I REGISTERS FILE (TTL): 74170, 74670, 74870, 74871, 74172

Un *Register File* è una matrice bidimensionale (*array*) di *MxN* memorie elementari di tipo D; può essere definito "*Banco di Registri*" e, in sostanza, è un insieme di *M* Registri (le colonne della matrice) ciascuno dei quali è in grado di memorizzare *N* bit (le righe della matrice); in questo assomiglia molto alle memorie RAM, nelle quali la lettura o la scrittura dei dati in esse contenuti può avvenire in qualsiasi ordine, puntando a caso una qualsiasi locazione, cioè, come si suol dire, "ad accesso casuale" (*Random Access*).

Per consentire la localizzazione del Registro in cui inserire, o da cui estrarre, un dato è necessario disporre di un certo numero di linee di controllo; alcune di esse servono per indicarne il numero (espresso in binario) e per questo sono dette *linee d'indirizzo*: se il numero dei Registri contenuti nel Banco è pari a *M*=8, saranno necessarie 3 linee, cioè sempre in quantità pari all'esponente della base 2 necessario per esprimere *M* (nel nostro esempio, appunto, 2^3=8). Stabilito l'indirizzo della locazione (Registro), è necessario attendere che il valore posto sulle linee di indirizzo sia stabile, e poi abilitare la scrittura o la lettura, con l'aiuto di una o più linee di sincronismo; di solito, come vedremo in altra puntata, alle memorie RAM ne basta una, il cui livello attivo (basso) abilita la scrittura, assicurando invece automaticamente la lettura delle celle puntate quando è disattiva (alta).

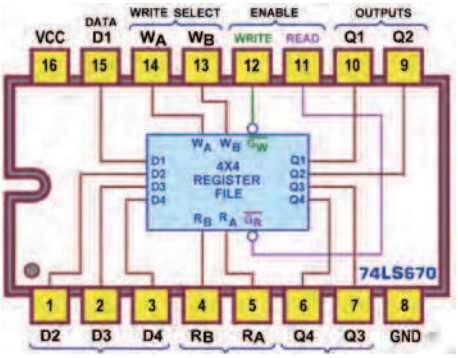


Figura 1: 4-by-4 Register File, 74LS670: Pin-out.

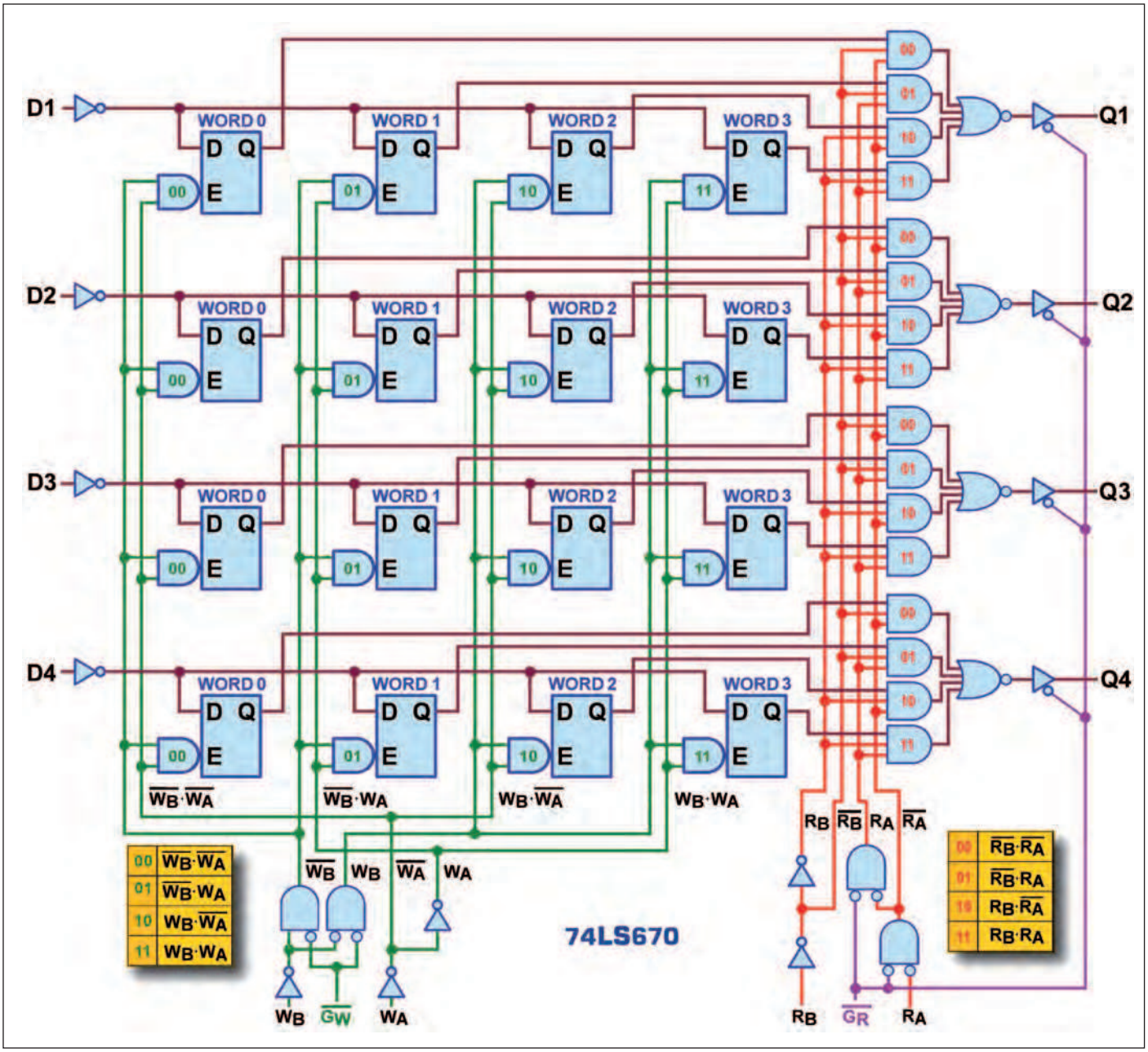


Figura 2: 4-by-4 Register File, 74LS670: Schema funzionale.

I *Registers File* sono, invece, componenti speciali molto versatili perché dispongono di linee separate e autonome (d'indirizzo e di abilitazione), sia per la scrittura che per la lettura; questo significa che ognuno degli *M* Registri può essere letto in qualsiasi momento anche se, contemporaneamente, è in atto una sua scrittura, cosa impossibile con le *RAMs*.
Per i dettagli funzionali è conveniente far riferimento alla manciata di componenti reali resi disponibili dalla famiglia TTL, a cominciare dal 74LS670, definito *4-by-4 Register File with 3-state outputs*, che permette la scrittura e la lettura simultanea di quattro diversi Registri; la **figura 1** mostra il suo pin-out, dal quale appare evidente la possibilità di controllare le 4 parole da 4 bit ciascuna con tre linee per la loro scrittura (*Write Select*, W_A e W_B, e *Write Enable*, G_W) distinte dalle tre linee necessarie per la loro lettura (*Read Select*, R_A e R_B, e *Read Enable*, G_R).
Lo schema funzionale **(figura 2)** è particolarmente impegnativo ma altrettanto utile per capire i segreti di questo dispositivo; intanto appare evidente che i 16 elementi di memoria utilizzati sono *D-Latch*, contrariamente a quelli usati nelle memorie RAM di tipo *D-Type*, cioè tali da "inseguire" in uscita il valore dell'ingresso D per tutto il tempo in cui la loro abilitazione E rimane a 1, memorizzando quello presente nell'istante corrispondente al fronte di discesa di E.
Si tratta dunque di memorie attive sui livelli (*Level Triggered*) del segnale di controllo (.. un'abilitazione, non un Clock) e il loro tipico "effetto trasparenza" obbliga, al fine di evitare cambiamenti di stato indesiderati, ad essere ben certi che l'ingresso D sia stabile e a pensare al segnale

di abilitazione E come ad un brevissimo impulso a 1 (sostanzialmente racchiuso dai suoi 2 fronti di salita e di discesa, così da essere ritenuti pressocché coincidenti), breve ma non troppo, per dare alla memoria almeno il tempo per commutare. Entrambe le condizioni sono facilmente assicurabili e comunque sono relative alla sola fase di scrittura; per ognuno dei *D-Latch* l'abilitazione E è assicurata dal prodotto logico (AND) di due segnali, ottenuti dalla decodifica del numero binario a 2 bit predisposto sulle linee d'indirizzo in scrittura, W_A e W_B, abilitata dal breve impulso attivo basso fornito su G_W.
La **figura 3a** mostra due varianti del dettaglio relativo al circuito di controllo della scrittura offerto da diversi datasheet e la **figura 3b** offre un paio di schemi, equivalenti ad entrambi i precedenti.
La parola a 4 bit da memorizzare è presentata ai quattro ingressi di dato (da D1 a D4), e finisce in parallelo sugli ingressi interni D di ciascun latch, ma ovviamente solo uno dei 4 Registri del Banco sarà autorizzato a riceverla. A questo provvede il decoder che, in funzione del valore corrente delle linee d'indirizzo W_A e W_B, farà in modo di aprire, forzando un 1 logico su entrambi i suoi ingressi, le sole porte AND ad esso relative.
In realtà le porte autorizzate dalla decodifica del codice binario rimangono aperte solo per il brevissimo istante durante il quale il segnale G_W, il vero sincronismo di scrittura, viene tenuto basso, rendendo la locazione indirizzata trasparente ai dati, e memorizzando quello presente nell'istante corrispondente al fronte di salita di G_W.
Da questo momento, per tutto il tempo durante il quale quest'ultimo rimane alto, le linee di ingresso dati sono inibite e il lo-

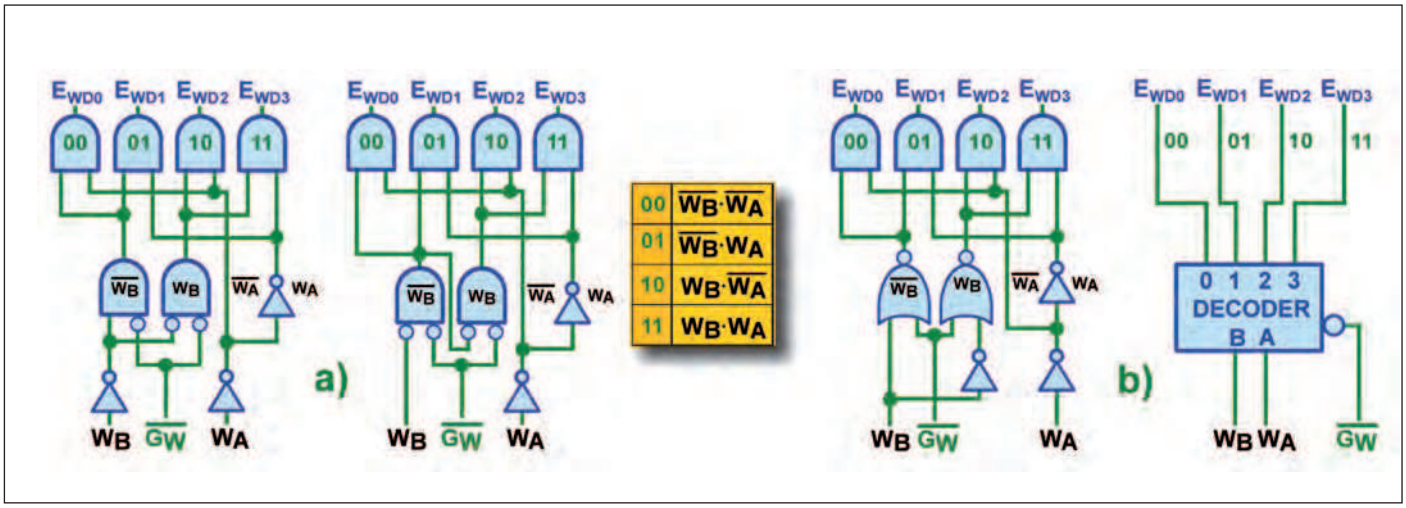


Figura 3: 4-by-4 Register File, 74LS670: Ingressi per la scrittura.

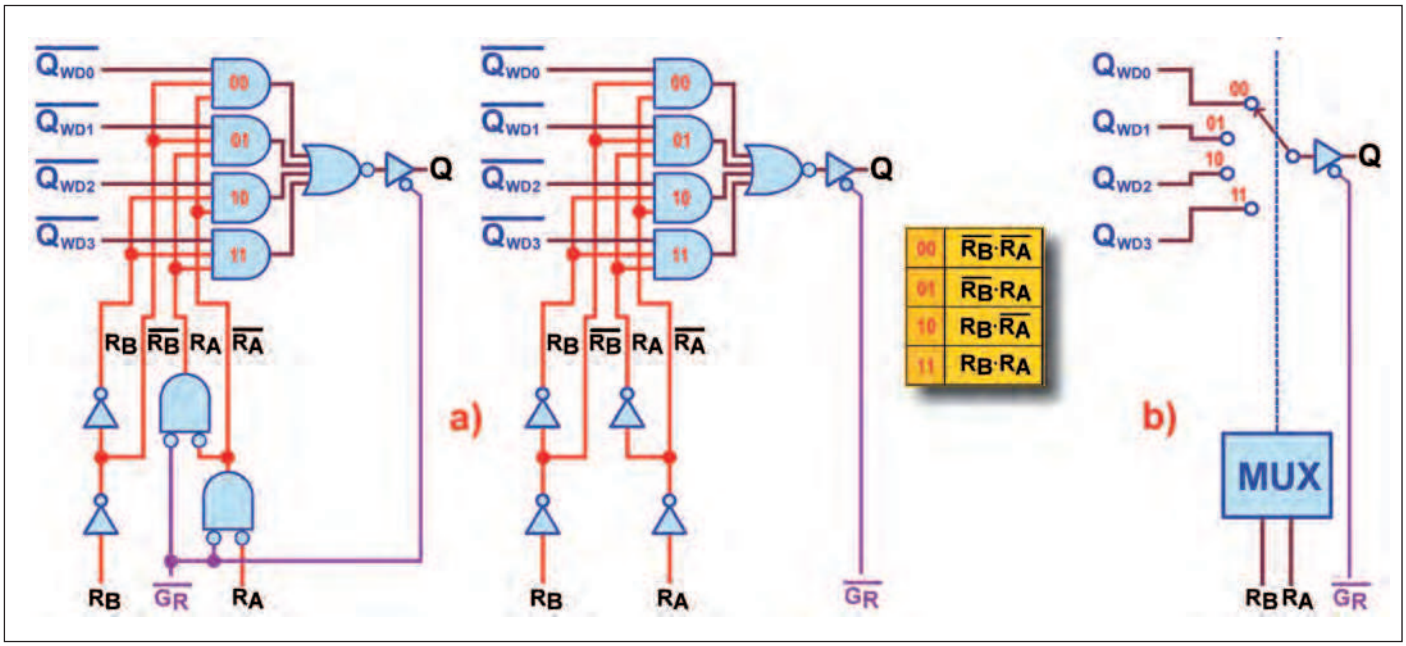


Figura 4: 4-by-4 Register File, 74LS670: Ingressi per la lettura.

Next Generation Intelligent LCDs

RISPARMIA TEMPO.
RISPARMIA DENARO.
RISPARMIA MANODOPERA.

Riduci significativamente i costi di sviluppo
Time-to-market ultra-veloce per le tue applicazioni

CODICE MIP 2808461

Per maggiori informazioni visita il sito **www.elettroshop.com/demmel** oppure chiama il numero **02/66504794**

ro livello non potrà causare alcun cambiamento alle informazioni memorizzate sulle uscite dei latches.
La parte circuitale destinata alla lettura è affidata ad un multiplexer "da 4 a 1" a quattro vie, una per ciascun gruppo di uscite interne Q dei 4 Latches interessati dai bit aventi lo stesso peso, nell'ambito dei Registri del Banco. La **figura 4a** mostra il dettaglio di una delle quattro vie, in due varianti estratte da diversi datasheet; vale la pena evidenziare la presenza in forma negata dei segnali coinvolti in ingresso (le uscite dei 4 Latch): ciò è dovuto alla presenza dell'inverter su ciascuna linea d'ingresso Dato D (vedi schema funzionale di **figura 2**) ed il corretto valore viene poi recuperato dall'inverter presente nella tipica forma AND-OR-NOT usata per il MUX. La **figura 4b** ne è una buona e semplice simulazione.
Tutti i bit presenti sulle uscite interne Q dei 16 latches sono contemporaneamente presenti sugli ingressi del Multiplexer, ma solo 4 saranno trasferiti sulle uscite esterne Q del *Register File*. Per la sua natura di "commutatore rotativo digitale", il MUX è chiamato a catturare simultaneamente tutti e 4 i bit del solo Registro puntato dalle sue linee di selezione, in sostanza costituite dalle linee d'indirizzo R_A e R_B dedicate esclusivamente alla lettura e del tutto distinte da quelle utilizzate nella fase di scrittura.
In realtà la parola contenuta nel Registro indirizzato appare sulle quattro uscite esterne solo quando anche il segnale di abilitazione alla lettura, G_R, viene tenuto basso, aprendo i buffers 3-state ad esse associati; se G_R è alto, le 4 uscite Q esterne sono inibite e passano allo stato di alta impedenza.
La **figura 5** sintetizza l'azione delle linee di controllo sui 4 latches del primo Registro del Banco, chiamato a memorizzare la parola 0 (Word0). Naturalmente, come detto, la fase di scrittura può avvenire mentre viene esercitata una lettura, anche da un Registro diverso da quello che si sta scrivendo; sebbene possa succedere che lo stesso Registro sia letto e

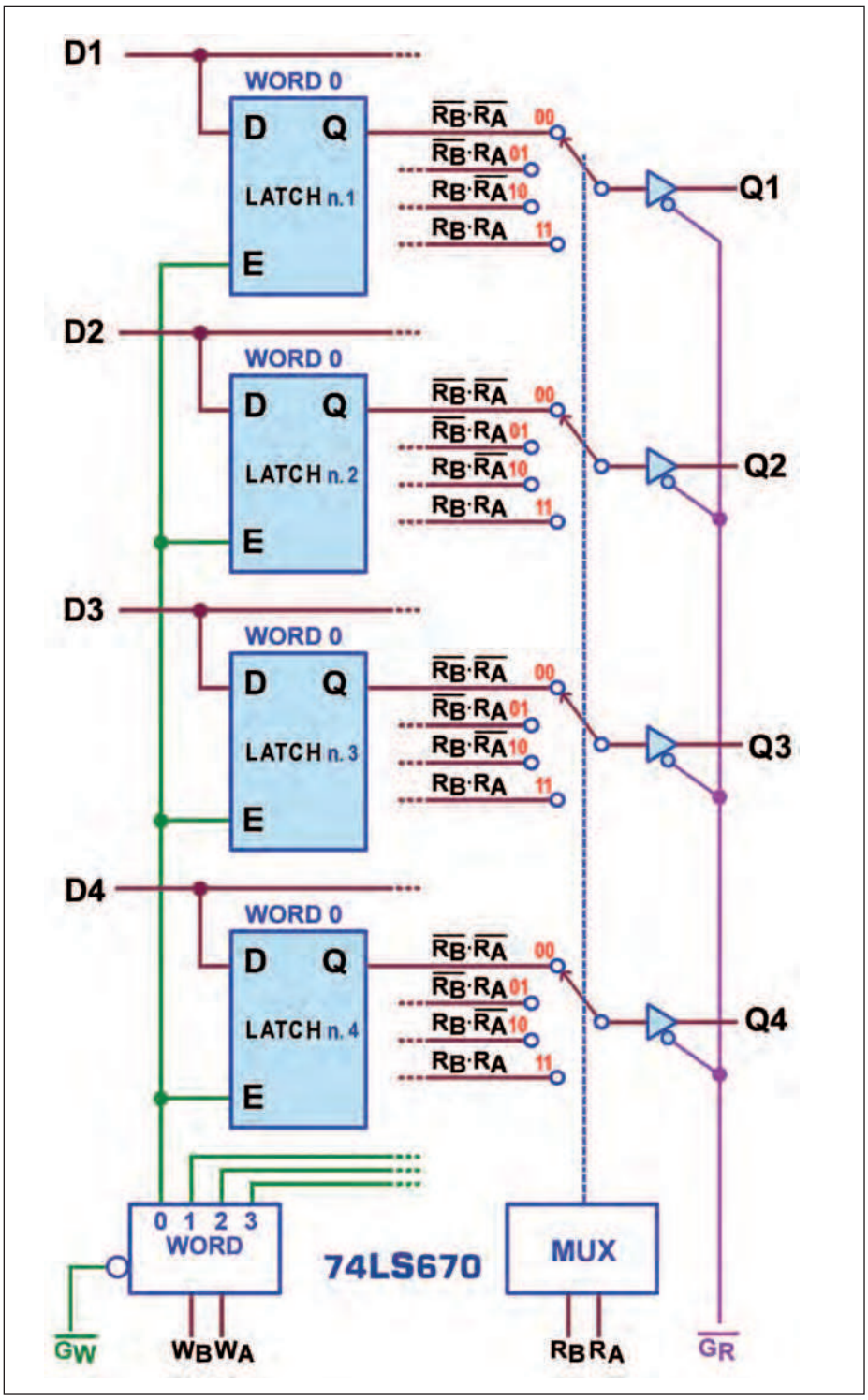


Figura 5: 4-by-4 Register File, 74LS670: Gestione del Registro Word0.

scritto nello stesso istante, se i dati proposti in ingresso non sono mantenuti stabili, prima e durante l'arrivo dell'impulso di scrittura G_W è consigliabile, a causa della "trasparenza" dei Latches, esercitare la lettura in tempi diversi.
La **figura 6** propone lo schema pratico mentre la **figura 7** mostra la Tabella di Verità, articolata in due parti per sottolineare l'attitudine di questo componente a essere scritto e letto contemporaneamente; in entrambe il pedice *n* (in questo componente uguale da 0 a 3) fa le veci delle linee d'indirizzo, non presenti in modo esplicito, ed associa i segnali che ne dispongono (D_n e Q_n) a tutti gli *n* bit del

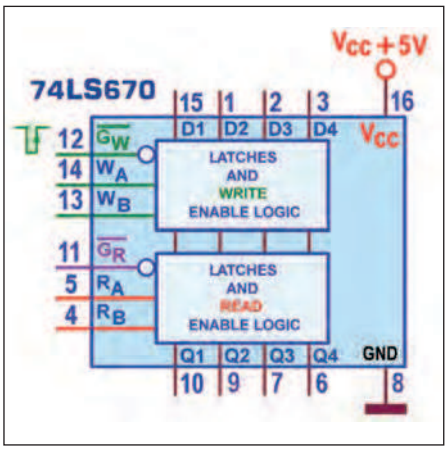


Figura 6: 4-by-4 Register File, 74LS670: Schema pratico.

Registro indirizzato. A corollario della descrizione notiamo che: **a**) nella fase di scrittura (*Write Mode Select*) viene sottolineata la necessità di mantenere stabili anche le linee d'indirizzo W_A e W_B mentre il segnale G_W è a basso livello; **b**) nella fase di lettura (*Read Mode Select*) si conferma che l'indirizzamento tramite R_A e R_B delle uscite interne Q può avvenire indipendentemente dall'azione esercitata dai segnali di abilitazione segnale G_W e G_R.
L'estrema versatilità di questo componente permette di coinvolgerne più di uno "in batteria", al fine di garantire la disponibilità di Registri contenenti parole più grandi, portando la loro dimensione a 8 o 12 o 16 bit, comunque multipla di 4, oppure per aumentare il numero di Registi nel Banco, portandolo a 8 o 12 o 16 bit, sempre multiplo di 4, o per l'una e l'altra opportunità.
In **figura 8** è disponibile il simbolo logico predisposto dallo standard IEEE; i segnali presenti nel blocco di controllo comune, posto in alto nel simbolo, sono quelli che si ritengono tali da influenzare tutti gli altri elementi presenti nel dispositivo: **a**) la prima notazione raccoglie i 2 ingressi di selezione A e B del decoder (quelli d'indirizzo in scrittura) in una parentesi graffa, che li fa corrispondere al simbolo 1A per identificare l'etichetta dei segnali (appartenenti alla serie dei blocchi sottostanti) da essi influenzati (cioè gli ingressi di dato D_n); il numero 0 e 1 indica il *peso* del bit nell'ambito della parola d'indirizzo, e la grafica $0/(2^n-1)$ alla destra della grafa non è una frazione ma indica l'intervallo previsto per gli ingressi sotto controllo, nel nostro esempio da 0 a 3; **b**)

WRITE MODE SELECT TABLE

INPUTS		OUTPUT	
D_n	$\overline{G_W}$	Q_n INTERNAL LATCHES	MODO
0	0	0	Scrittura del Dato
1	0	1	
X	1	non cambia	Memorizzazione

READ MODE SELECT TABLE

INPUTS		OUTPUT	
Q_n INTERNAL LATCHES	$\overline{G_R}$	Q_n	MODO
0	0	0	Lettura del Dato
1	0	1	
X	1	Hi-Z	Alta Impedenza

Figura 7: 4-by-4 Register File, 74LS670: Tabella di Verità.

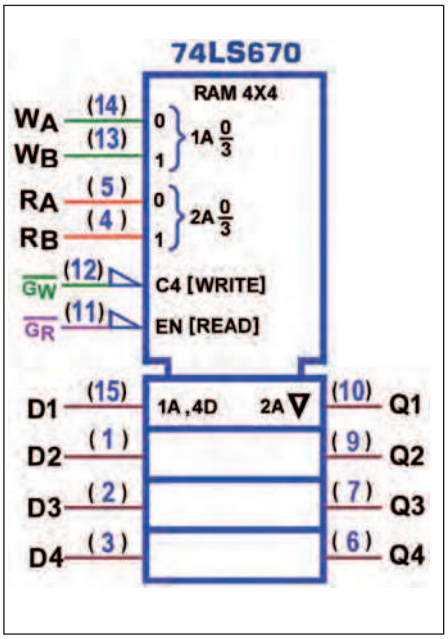


Figura 8: 4-by-4 Register File, 74LS670: Simbolo logico ANSI/IEEE Std. 91-1984.

la seconda notazione è analoga alla prima ma rappresenta ora i 2 ingressi di selezione A e B del multiplexer (quelli d'indirizzo in lettura): il simbolo 2A identifica l'etichetta dei segnali da essi influenzati (ora le uscite Q_n), mentre la "frazione" alla destra della graffa ha la stessa funzione di prima; **c**) poiché i segnali di abilitazione, indicati con C4 e con EN, sono unici per tutte le quattro sezioni sottostanti vengono inclusi nella metà superiore, altrimenti ciascuna sezione avrebbe i propri. La parte inferiore del simbolo è composta da 4 blocchi, uno per *D latch*, identici tra loro e per i quali solo il primo deve essere redatto con dettaglio. Nel nostro caso esso contiene: **a**) l'etichetta 1A, 4D chiaramente riferita ai Dati in ingresso, con davanti il numero 1A per far capire a quale segnale del blocco di controllo è da ritenersi correlata, quello del decoder appunto; **b**) l'etichetta 2A, in prossimità delle uscite Q, per correlarle al blocco di controllo multiplexer; c) un triangolo rivolto verso il basso, per indicare che le uscite sono 3-state e che l'azione del comando in relazione con esse sarà quella di abilitare la disponibilità dei dati, oppure di disabilitarle in blocco (cioè di porle nello stato di alta impedenza).
La potenza massima dissipata dal 74LS640 è di 250 mW; la corrente erogata tipica in uscita (I_{OH}, con Q=1) e di 2,6 mA mentre quella assorbita massima in uscita (I_{OL}, con Q=0) è di ben 24 mA; in regime dinamico (con carico di 45pF/667ohm) i ritardi di propagazione massimi, per entrambe le transizioni t_{PLH} e t_{PHL} e per ogni ingresso (dati, selezione in lettura e abilitazione in scrittura) rispetto ad ogni uscita Q, sono mediamente di 45 ns; il valore

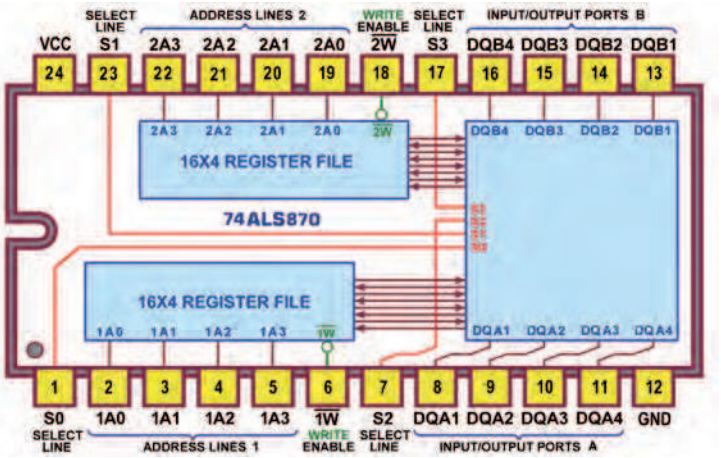


Figura 9: Dual 16-by-4 Register File, 74LS870: Pin-out.

minimo di Setup Time=t_S=10 ns e quello di Hold Time=t_H=15 ns. In questo caso viene fornito anche il tempo di latch (t_{LATCH}=25 ns, massimo), cioè il tempo concesso all'uscita interna Q del latch per assumere il valore di nuovi dati. Esso è importante solo quando si cerca di leggere da una locazione immediatamente dopo la sua scrittura.

Del Banco di Registri 74LS670 (*4-By-4 Register Files*) esiste una versione del tutto identica, il 74LS170, ma con uscite di tipo Open-Collector. Tutte le considerazioni appena fatte rimangono dunque valide, compresi gli schemi, ovviamente con le piccole differenze dovute alla nuova natura delle uscite: per esempio nello schema funzionale di **figura 2** mancheranno i 4 buffer 3-state e il segnale di abilitazione alla lettura G_R si limiterà a tenere aperte le due AND che governano l'indirizzo R_A.

Anche il simbolo logico predisposto dallo standard IEEE dovrà contenere un dettaglio differente: nella parte inferiore del simbolo, quella dei 4 blocchi che rappresentano i *D latch* del dispositivo, sull'estrema destra del primo apparirà ora un piccolo rombo, destinato a giustificare le uscite open collector, al posto del "*triangolo rivolto verso il basso*" tipico delle uscite sono 3-state.

La potenza massima dissipata dal 74LS170 è di 200 mW e la corrente assorbita in uscita (I_{OL}, con Q=0) può essere al massimo di 8 mA; il comportamento in regime dinamico, con carico di 15pF/2kohm, è di fatto lo stesso di quello con uscite 3-state.

La serie TTL, per altro nella sua versione avanzata, ALS, *Advanced Low Power Schottky*, prevede un altro componente di questo tipo, molto sofisticato: si tratta del 74ALS870 (un *Dual 16-By-4 Register File with 3-state outputs*) dotato 2 Banchi, ciascuno organizzato con 16 Registri a 4 bit; ogni Registro dispone delle proprie 4 linee d'indirizzo (da A0 a A3, decodificate da un decoder interno da 4 a 16, funzionalmente simile al 74LS154, per poter puntare *ad accesso casuale* le 2^4=16 locazioni) e della propria linea di abilitazione alla scrittura (W, *Write Enable*, attiva bassa); la **figura 9** mostra il suo pinout. Questo dispositivo è poi dotato di due porte bidirezionali di input/output a 4 bit (DQA1-DQA4 e DQB1-DQB4) e di una complicata rete combinatoria che ha il compito di organizzare il trasferimento dell'informazione in transito su di esse dalle uscite interne o verso gli ingressi interni dei 2 Banchi di Registri, in funzione del valore assunto da ulteriori 4 linee di controllo, da S0 a S3.

REGISTERS FILE SELECT			INPUT/OUTPUT SELECT		
S0	S1	Selezione del Registro coinvolto	S2	S3	Flusso Dati sui Bus
0	0	Registro1 verso A, Registro1 verso B	0	0	Porta A in OUTPUT Porta B in OUTPUT
1	0	Registro2 verso A, Registro1 verso B			
0	1	Registro1 verso A, Registro2 verso B			
1	1	Registro2 verso A, Registro2 verso B			
0	0	A verso Registro1, Registro1 verso B	1	0	Porta A in INPUT Porta B in OUTPUT
1	0	A verso Registro2, Registro1 verso B			
0	1	A verso Registro1, Registro2 verso B			
1	1	A verso Registro2, Registro2 verso B			
0	0	Registro1 verso A, B verso Registro1	0	1	Porta A in OUTPUT Porta B in INPUT
1	0	Registro2 verso A, B verso Registro1			
0	1	Registro1 verso A, B verso Registro2			
1	1	Registro2 verso A, B verso Registro2			
0	0	B verso Registro1	1	1	Porta A in INPUT Porta B in INPUT
1	0	A verso Registro2, B verso Registro1			
0	1	A verso Registro1, B verso Registro2			
1	1	B verso Registro2			

Figura 10: Dual 16-by-4 Register File, 74LS870: Tabella di Verità.

In sostanza, il componente è progettato specificamente per Architetture Multibus: di fatto le due porte sono collegate a due distinti bus, e sono in grado di pilotarli direttamente, disponendo di uscite bufferizzate di tipo 3-state; la destinazione di ciascuna possibile *parola a 4 bit* dipende dal valore del codice binario a 4 bit presente sulle linee di selezione: **a**) potrà essere trasmessa in uscita su uno dei 2 *bus dati*, dopo essere stata letta da uno dei Registri; **b**) potrà essere ricevuta da uno dei *bus dati* e poi scritta in uno dei Registri; **c**) potrà essere ricevuta in ingresso da uno dei due bus A e spedita in uscita sull'altro bus, e viceversa.

Per la numerosa quantità di porte logiche "di contorno" coinvolte, in aggiunta ai 32 Latches di tipo D, necessari per i 2 Registers File, lo schema funzionale è decisamente troppo complesso; anche i datasheet si limitano a fornirne una versione molto schematica, che rischia di rimanere piuttosto oscura e di difficile comprensione: in questo caso il suo disegno diventa controproducente e per sintetizzarne il funzionamento risulta più convincente la Tabella di Verità, proposta in **figura 10**.

Da essa si capisce che due delle linee di selezione, S0 e S1, hanno il compito di stabilire quale delle 2 Porte di I/O e a quale dei 2 Banchi di Registri appartiene la locazione indirizzata che verrà messa in contatto con la porta, al fine di scambiarsi il dato a 4 bit: il flusso dei dati potrà avvenire in quattro modi diversi, tante quante sono le codifiche binarie possibi-

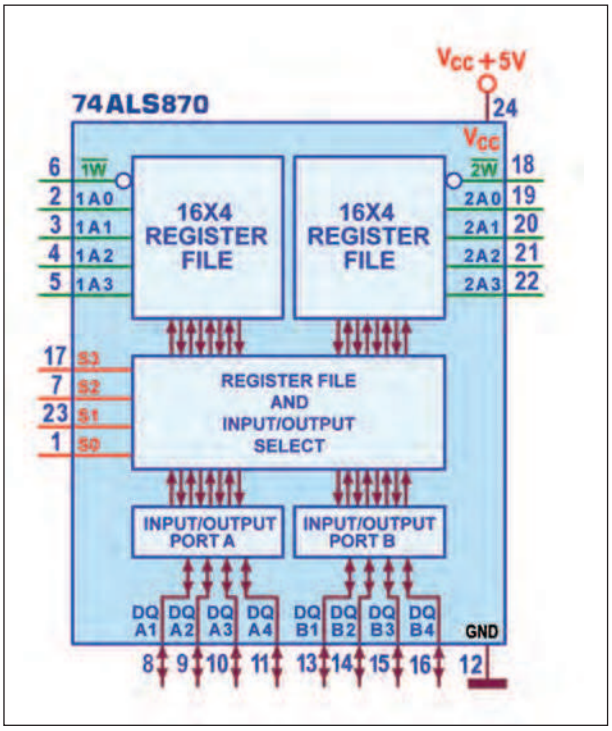


Figura 11: Dual 16-by-4 Register File, 74LS870: Schema Pratico.

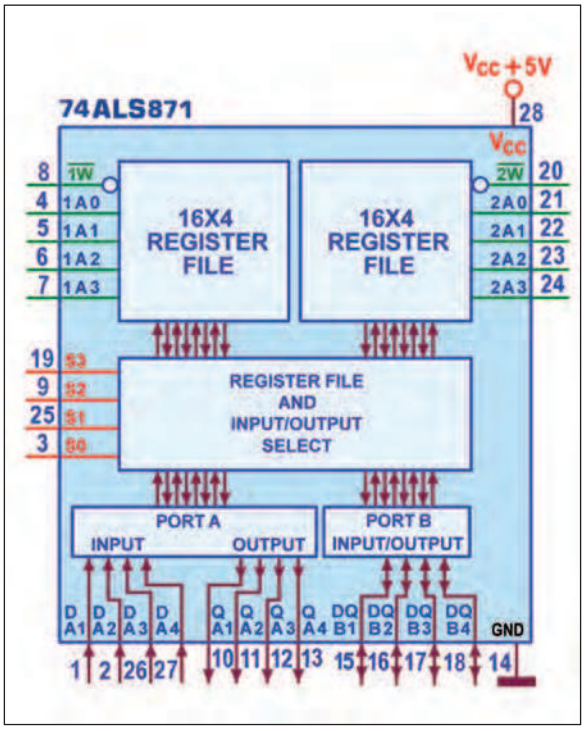


Figura 13: Dual 16-by-4 Register File, 74LS871: Schema Pratico.

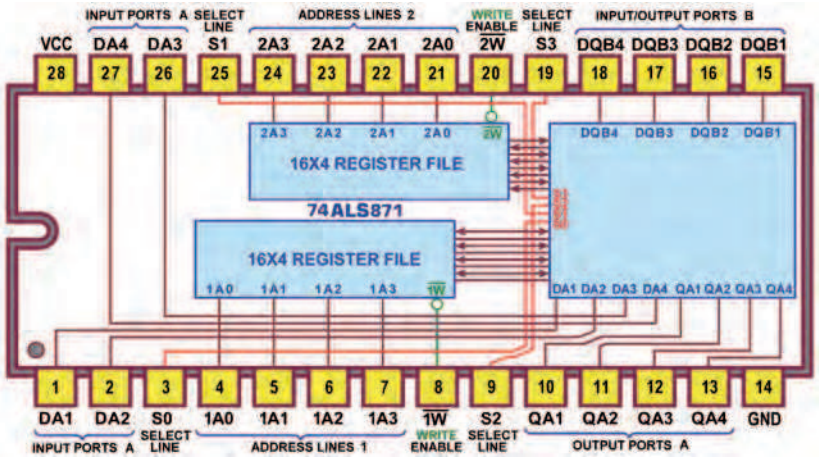


Figura 12: Dual 16-by-4 Register File, 74LS871: Pin-out.

li sulle 2 linee S0 e S1. La linea S2 stabilisce se la *porta A* di I/O è programmata in ingresso (S2=1) o in uscita (S2=0) e la linea S3 fa lo stesso servizio per la *porta B* di I/O, programmata in ingresso (S3=1) o in uscita (S3=0).

Osservando le ultime quattro righe della tabella, la prima e l'ultima sembrano incomplete; a rigor di logica la combinazione S0,S1=00 dovrebbe selezionare "*A verso Registro1, B verso Registro1*" ma la cosa è fisicamente intollerabile; la stessa considerazione vale per la combinazione S0,S1=11 che dovrebbe orientare entrambe le *porte A e B* verso lo stesso *Registro2*. Per evitare questo conflitto in scrittura, tipico solo durante la modalità a doppio ingresso, le specifiche del dispositivo prevedono che la Porta d'ingresso B abbia la priorità sulla Porta d'ingresso A.

In **figura 11** è disponibile lo schema pratico, utile per l'impiego diretto nel contesto di un progetto.

La potenza massima dissipata dal 74ALS870 è di 550 mW e la corrente massima in uscita è tipicamente pari a 2,6 mA (erogata, I_{OH} con Q=1) e di 24 mA (assorbita, I_{OL} con Q=0), decisamente degna di nota; il valore minimo di Setup Time prima del sincronismo di scrittura, t_S, va da 5 ns (indirizzi) a 15 ns (dati), passando per i 12 ns (linee di selezione); quello di Hold Time, t_H, è sostanzialmente nullo; i tempi d'accesso massimi, per ogni ingresso (dati, selezione, indirizzi) rispetto ad ogni uscita DQ, sono mediamente di 16 ns.

Il componente 74ALS871 è una versione a 28 pin del precedente, funzionalmente identico ma con 4 piedini in più, per consentire una migliore disponibilità dei segnali per la porta A, ora gestibile in modo unidirezionale, separatamente per l'ingresso (DA1 - DA4) e per l'uscita (QA1 - QA4); l'altra porta B rimane bidirezionale (DQB1 -DQB4) e comunque prioritaria sulla porta A, quando entrambe sono programmate in ingresso. La **figura 12** mostra il suo pin-out e la **figura 13** il suo schema pratico.

L'ultimo componente della categoria è ancora più particolare: si tratta del 74LS172, definito *16 bit Multiple Port Register File with 3-state outputs*. Esso dispone di due sezioni identiche ciascuna in grado di gestire le medesime 8 parole da 2 bit della piccola memoria, simultaneamente sia in lettura che in scrittura; ogni sezione dispone infatti di un proprio apparato separato per la lettura e per la scrittura, indipendente ed autonomo rispetto ad ogni altro.

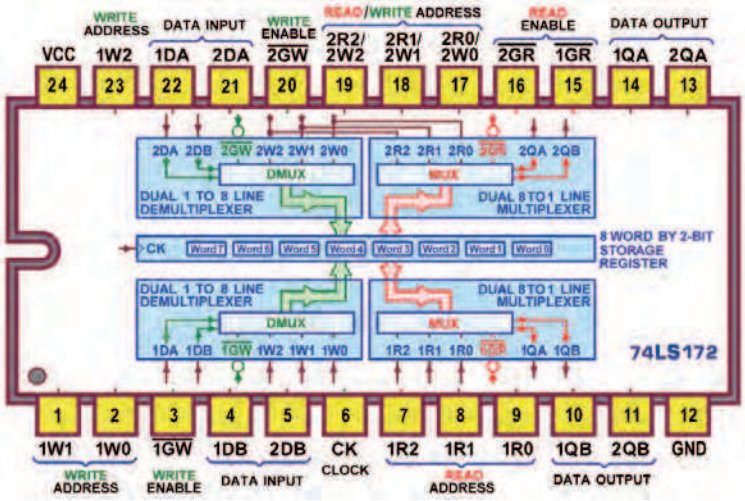


Figura 14: 16 bit Multiple Port Register File, 74LS172: Pin-out.

La **figura 14** mostra il suo complesso pin-out. La presenza di una linea di Clock sottolinea che, in questo caso, si tratta a tutti gli effetti di un dispositivo assimilabile ad una *Ram 8x2*, realizzato cioè con 16 Flop-Flop D-Type. Questo sincronismo è in comune ad entrambe le sezioni e provvede a memorizzare i nuovi dati nelle locazioni indirizzate sul suo fronte di salita, se contemporaneamente la rispettiva linea di abilitazione alla scrittura, 1GW e/o 2GW, è bassa.

La **figura 15** espande in dettaglio una delle due sezioni, per altro uguali tra loro, che concorrono sulle 16 locazioni del Banco degli 8 Registri a 2 bit, evidenziandone il sofisticato funzionamento. Ci riferiamo, ad esempio, alla prima delle due sezioni per cui tutti i segnali avranno il prefisso "1" (va da se che ogni descrizione vale anche per l'altra sezione, sostituendo "2" al posto di detto prefisso): a) i dati in ingresso esterni, 1DA, 1DB, sono forniti alle due linee di input di un *doppio demultiplexer "da 1 a 8"* e da questo presentati sulle linee interne D dei 2 Flip-Flops corrispondenti ai due bit della parola indirizzata dalle linee 1W0,1W1,1W2; se l'abilitazione alla scrittura 1GW è bassa, questa informazione viene memorizzata sulla transizione positiva dell'ingresso di clock; b) se l'abilitazione alla lettura 1GR è bassa, con l'aiuto di un *doppio multiplexer "da 8 a 1"*, il contenuto di una delle 8 parole del Banco di Registri potrà essere letto (anche in concomitanza con l'eventuale fase di scrittura) sulle uscite 1QA,1QB, che assumeranno i livelli logici memorizzati sulle linee d'uscita interne Q dei 2 Flip-Flops corrispondenti ai due bit della parola indirizzata dalle linee 1R0,1R1,1R2; se l'abilitazione alla lettura 1GR è alta, le uscite associate alle locazioni indirizzate si trovano nello stato di alta impedenza.

Poiché le due sezioni sono indipendenti, può succedere che entrambe stiano puntando (rispettivamente con 1W0,1W1,1W2 e con 2W0,2W1,2W2) la medesima word (Registro a 2 bit), cercando di memorizzare in essa l'informazione presente sui propri ingressi di dato (rispettivamente 1DA,1DB e 2DA,2DB); se si verifica che l'informazione non è la stessa per entrambe le sezioni (cioè 1DA diverso da 2DA e/o 1DB diverso da 2DB) in ogni bit da memorizzare sarà predominante il dato a livello basso.

Da notare che, per evidenti motivi di spazio (il chip ha solo 24 piedini...), nella seconda sezione le linee d'indirizzo utilizzate per la scrittura e per la lettura sono le stesse.

Per ogni ingresso, il valore minimo t_S di Setup Time è mediamente di 35 ns e quello t_H di Hold Time è nullo; il tempo d'accesso tipico è di 15 ns in lettura e di 33 ns in scrittura. ❑

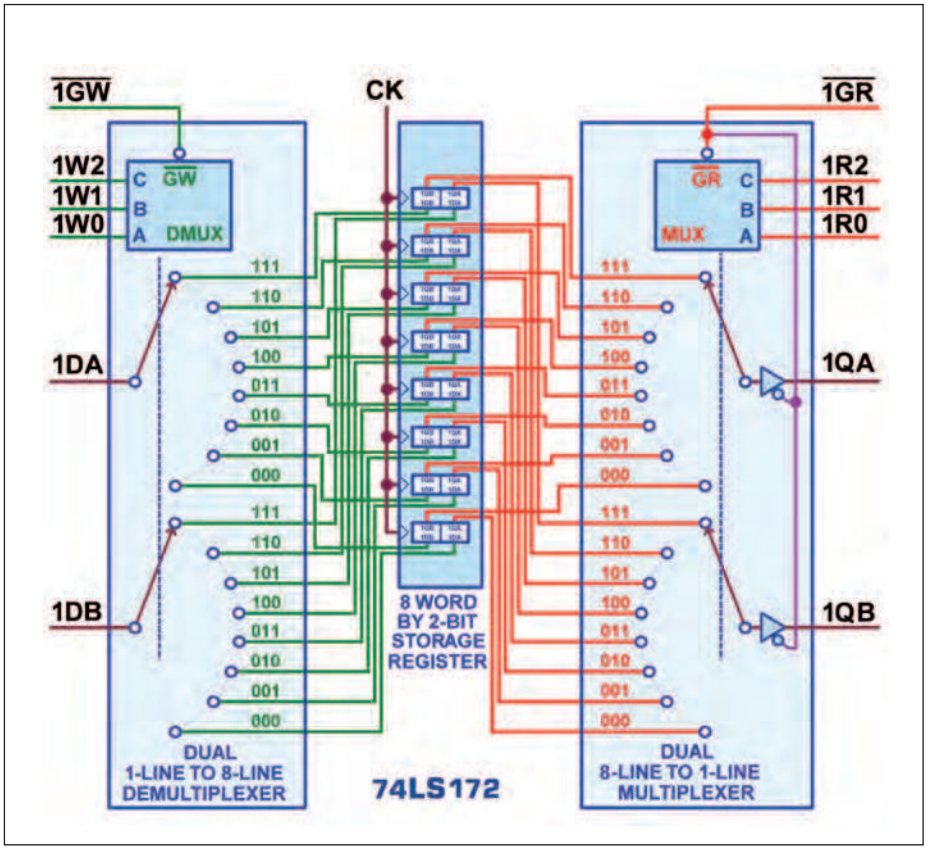


Figura 15: 16 bit Multiple Port Register File, 74LS172: Dettaglio Funzionale.

CODICE MIP 280485

il tuo appuntamento estivo ti aspetta da giugno in edicola NON MANCARE!!!

se non la trovi ordinala qui: www.farelettronica.com/c2011

CODICE MIP 2808465

MINI *ricetrasmettitore* MORSE

Un semplice RTX da costruire nei ritagli di tempo che vi permetterà di esercitarvi nell'antica arte delle trasmissioni in codice Morse

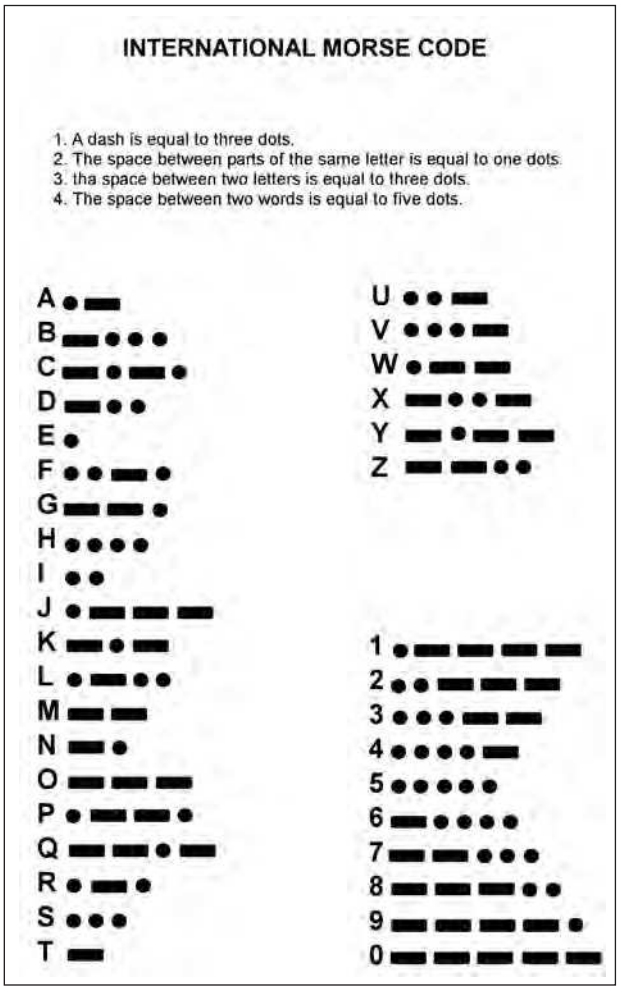


Figura 1: tabella codice Morse.

Chi dice che il codice Morse sia un sistema arcaico di trasmettere e ricevere informazioni, ormai non più utilizzato, può essere facilmente smentito: è sufficiente sintonizzare una radio per onde corte sulle bande radioamatoriali in HF e scoprire, in particolar modo la sera, una "sinfonia", quasi ininterrotta, di segnali codificati in Morse. Certo, ai giorni nostri, di sistemi per trasmettere messaggi "nell'etere" ce ne sono un'infinità, ma vi posso assicurare che il Morse è ancora il sistema che richiede la tecnologia più semplice per essere implementata, garantendo, allo stesso tempo, collegamenti a lungo raggio anche con deboli potenze. Per contro, diciamo che è necessaria un'applicazione particolare per imparare a ricevere e poi a trasmettere, come pure, una volta appreso, occorre continuare a esercitarsi in maniera regolare e continua.

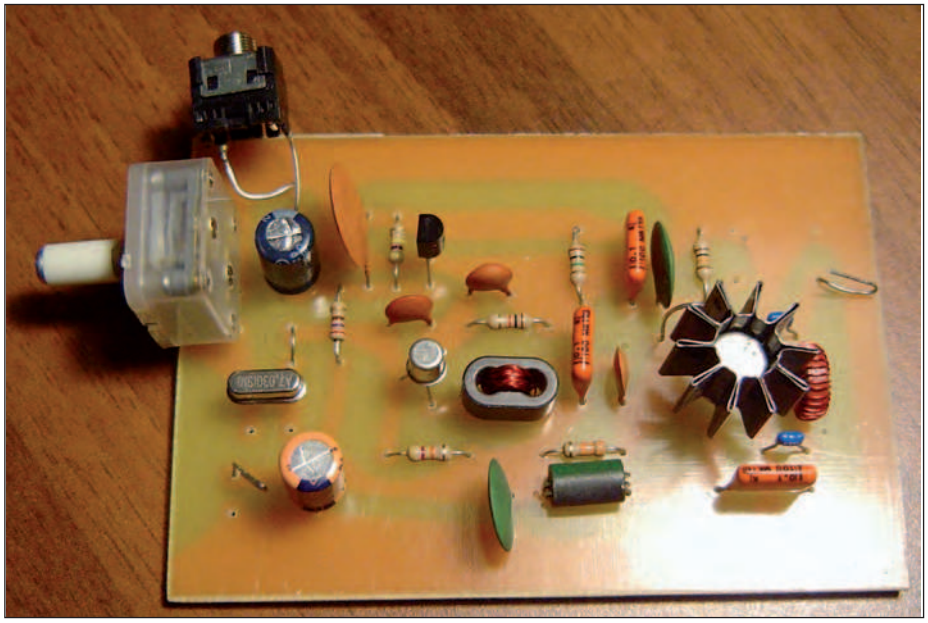
IL CODICE MORSE

Questo tipo particolare di trasmissione prende il nome dal suo ideatore, Samuel Morse, che iniziò a interessarsi di telegrafia nel 1832. Nel 1835 aveva progettato un primo sistema di relè che fu poi presentato pubblicamente nel 1837 e migliorato nell'anno successivo. Nel 1840 brevettò il telegrafo e la prima linea, entrare ufficialmente in funzione, fu tra Baltimore e Washington il 24 maggio del 1844. Il codice Morse, in origine, prevedeva la trasmissione di combinazioni di punti e linee dove ogni combinazione rappresentava in realtà un numero cui, a sua volta, era associata una determinata parola.Samuel Morse, nella realizzazione pratica del Telegrafo si avvalse dell'aiuto di Alfred Vail, il quale sviluppò, in seguito, un sistema in cui le sequenze di punti e linee rappresentavano singole lettere o numeri. Il codice Morse divenne in breve tempo il sistema standard per la codifica e le trasmissioni delle informazioni sia via cavo sia, in seguito, via radio, fino al 1999 rimase ufficialmente il sistema di comunicazione internazionale per i collegamenti radio marittimi. Considerando la com-

plessità di codificare in sequenze di punti e linee, parole e/o frasi molto lunghe, vennero escogitati alcuni codici commerciali in modo da raccogliere intere frasi in gruppi di poche lettere che venivano poi inviati come singole parole. Tra questi codici ricordo, soprattutto, il codice Q normalmente utilizzato nelle trasmissioni radioamatoriali e, in generale, dagli operatori di telegrafia per trasmettere informazioni di servizio come la qualità del collegamento, il cambio di frequenza o la numerazione dei telegrammi.

SCHEMA ELETTRICO

Premetto che la semplicità del circuito è tale che per il buon funzionamento del nostro piccolo ricetrasmettitore è necessaria un'adeguata antenna e una buona presa di terra. Non aspettatevi di captare trasmissioni in codice Morse con il consueto filo lungo un paio di metri collegato al posto di una vera antenna, al limite, costruendo due esemplari e rimanendo abbastanza vicini, con una piccola antenna di "fortuna" sarà possibile stabilire dei collegamenti a breve distanza con qualche amico/compagno che abita nelle vicinanze. Ricordatevi, comunque, che trasmettere sulle bande radio amatoriali è consentito solo se si è dotati di apposita patente conseguita presso il ministero delle comunicazioni. Ma veniamo allo schema elettrico: il circuito è composto di un oscillatore quarzato impiegante il transistor TR1 (2N706 o equivalente) come elemento attivo e montato in configurazione Colpitts, la reazione positiva è dovuta al partitore capacitivo composto da C4 e C5. Il segnale generato è disponibile su trasformatore al larga banda T1 che accoppia lo stadio oscillante con il secondo stadio basato sul transistor TR2. Questo stadio è studiato per funzionare sia come amplificatore di potenza in classe A, che miscelatore e rivelatore a conversione diretta.
Il duplice funzionamento dipende dal tasto telegrafico che, quando è chiuso, cortocircuita la resistenza R5 e il condensatore C9, permettendo al transistor



QRA	D:	Qual è il nome della tua stazione?
	R:	Il mio nominativo è...
QRB	D:	A che distanza approssimativa ti trovi dalla mia stazione?
	R:	La distanza tra la mia stazione e la tua è di circa... km.
QRG	D:	Qual è la mia frequenza esatta (o la frequenza di...)?
	R:	La tua frequenza esatta (o quella di...) è... kHz (o MHz).
QRH	D:	La mia frequenza varia?
	R:	La tua frequenza varia.
QRI	D:	Qual è la tonalità della mia emissione?
	R:	La tua tonalità è... .
QRK	D:	Qual è la comprensibilità dei miei segnali (o dei segnali di...)?
	R:	La comprensibilità dei tuoi segnali (o quelli di...) è...
QRL	D:	Sei occupato?
	R:	Sono occupato, questa frequenza è occupata.
QRM	D:	Sei disturbato da emittenti limitrofe (interferenze)?
	R:	Sono disturbato da interferenze.
QRN	D:	Sei disturbato da disturbi atmosferici?
	R:	Sono disturbato da disturbi atmosferici.
QRO	D:	Devo aumentare la potenza di emissione?
	R:	Aumenta(o) la potenza di trasmissione.
QRP	D:	Devo diminuire la potenza di emissione?
	R:	Diminuisci(o) la potenza di trasmissione.

Figura 2: estratto codice Q.

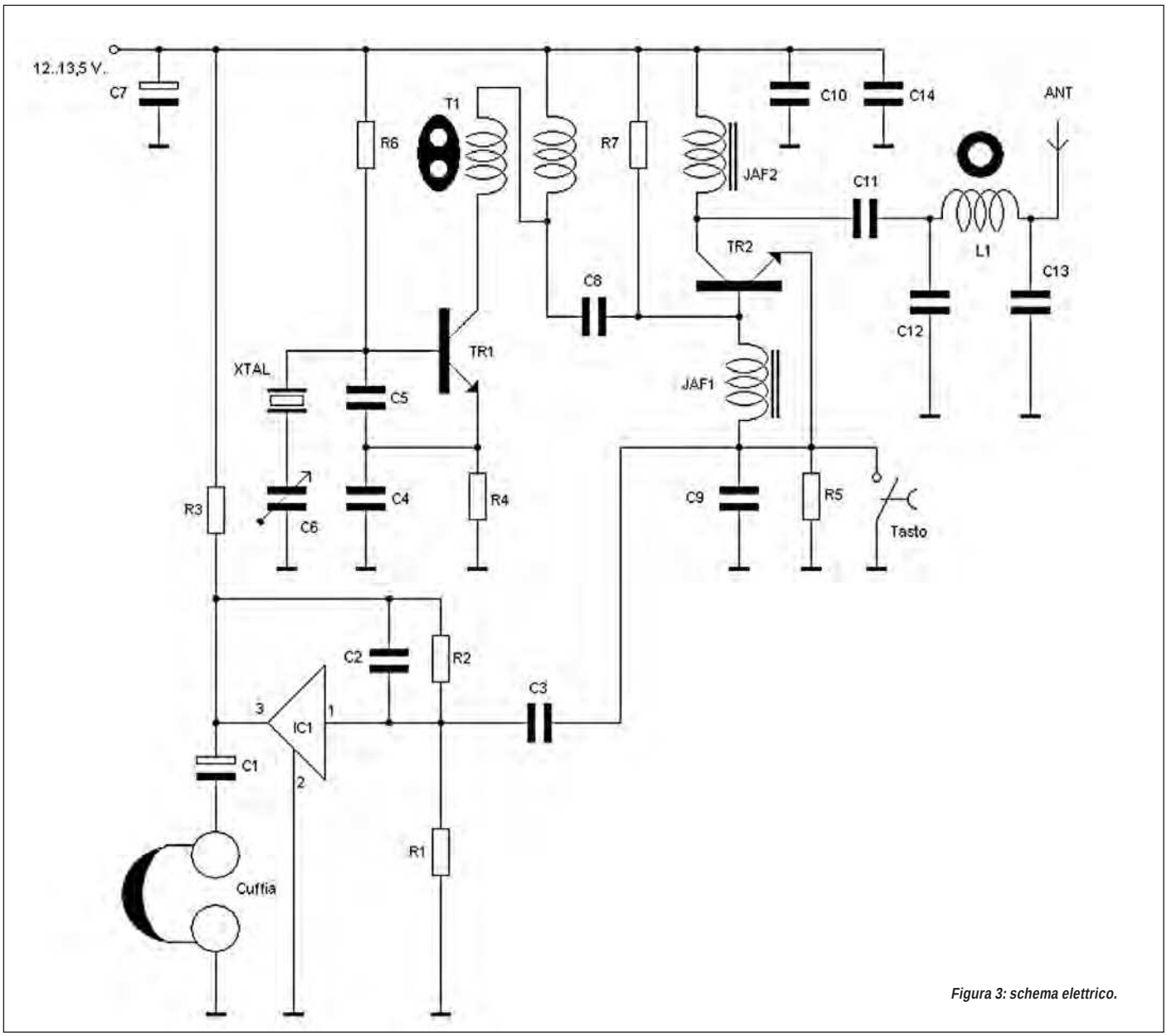
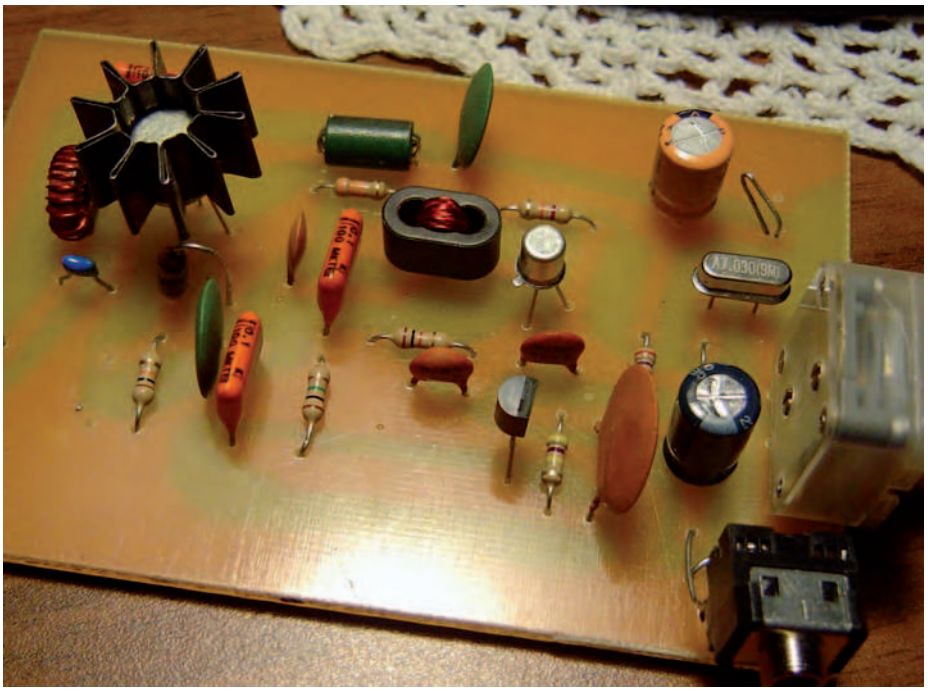


Figura 3: schema elettrico.

LISTA COMPONENTI

R1	1 Mohm ¼ w	C1	47uf 25v elettrolitico	C11	100nf pol.
R2	470 kohm ¼ w	C2	270pf ceramico	C12	820 pf ceramico
R3	270 ohm ¼ w	C3	100nf pol.	C13	820 pf ceramico
R4	1 kohm ¼ w	C4	560pf ceramico	C14	100nf pol.
R5	10 kohm ¼ w	C5	560pf ceramico	TR1	2N706 o equiv.
R6	47 kohm ¼ w	C6	10-100 pf cond. Variabile	TR2	2N3725 o equiv.
R7	33 kohm ¼ w	C7	47uf 25v elettrolitico	XTAL	7.030 Khz
		C8	47nf ceramico	T1,L1	Vedi testo
		C9	22nf ceramico	JAF1	100 uh
		C10	47nf ceramico	JAF2	VK200



PER *approfondire...*

http://it.wikipedia.org/wiki/Codice_Morse

http://it.wikipedia.org/wiki/Codice_Q

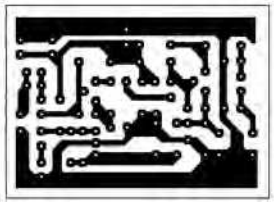
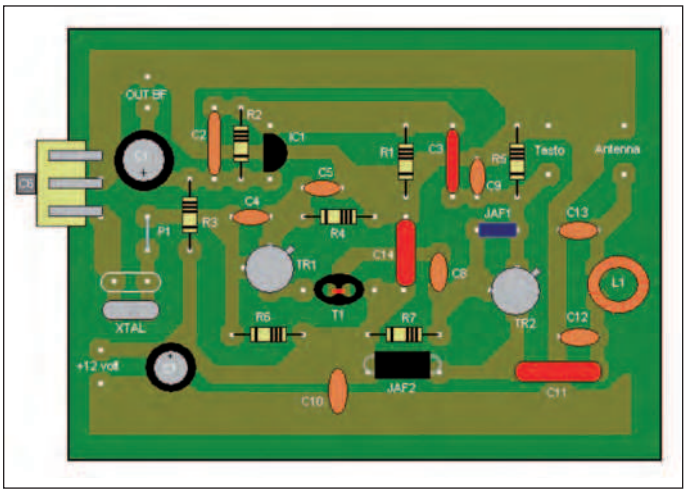


Figura 4: il circuito stampato

Figura 5: piano di montaggio



di amplificare il segnale prodotto dall'oscillatore e rendendolo disponibile in uscita dal filtro a pi greco formato da C12, C13 e L1. Se il tasto è aperto, la polarizzazione dell'emettitore fa funzionare il transistor come miscelatore, rendendo disponibile ai capi di C9, R5 il segnale audio ottenuto per battimento tra la portante generata dall'oscillatore con l'eventuale stazione morse ricevuta in antenna. Completa il circuito, un piccolo, quanto insolito, amplificatore a bassa frequenza, ottenuto utilizzando un circuito integrato impiegato normalmente come regolatore di tensione ma che nel nostro caso è usato per rendere udibile in una comune cuffia a bassa impedenza i segnali ricevuti.

MONTAGGIO E TARATURA

Il montaggio del mini RTX è abbastanza semplice, in **figura 4** è possibile vedere il circuito stampato mentre in **figura 5** abbiamo il piano di montaggio con la disposizione dei componenti. Gli elementi che vanno costruiti sono i due induttori T1 e L1,

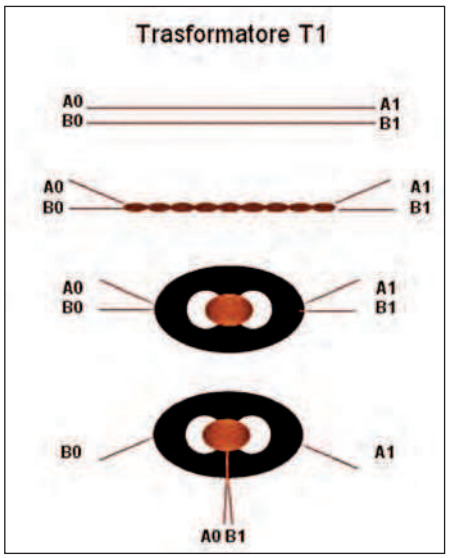


Figura 6: trasformatore T1.

che vanno realizzati nel seguente modo:

T1:prendete due spezzoni di filo di rame smaltato da 0,3 mm lunghi circa 50 cm e attorcigliateli in maniera simmetrica, utilizzate il filo così preparato per avvolgere in una ferrite binoculare tipo Amidon "BN 43-1502", 12-13 spire, il numero non è critico. Allargate i fili componenti i due estremi del trasformatore e con l'ausilio di un tester marcate i 4 terminali; unite A0 e B1, ottenendo così la presa che sarà collegata a C8 **(figura 6)**.

L1: avvolgere 17 spire di filo di rame smaltato su un nucleo toroidale T37-2.

Una volta montati tutti i componenti sulla basetta, alimentate il circuito e verificate con un frequenzimetro il regolare funzionamento dell'oscillatore. Il quarzo utilizzato nel prototipo e il filtro a pi greco sono per la banda radioamatoriale dei 40 metri, ma, naturalmente, è possibile, previo ricalcolo di C12, C13 e L1, configurare il circuito su frequenze diverse. Tramite il condensatore variabile C6 è possibile regolare leggermente la frequenze generata dall'oscillatore, in maniera da variare la nota di battimento durante la ricezione dei segnali telegrafici, aggiungo che inserendo in parallelo al quarzo un ulteriore quarzo di identica frequenza e sostituendo il ponticello P1 con un'induttanza di 15-20 uH (fate qualche tentativo) è possibile avere una variazione di frequenza molto più ampia. ❑

CODICE MIP 2804825

Radiocomando a 8 canali

il TRASMETTITORE

Di radiocomandi ne sono stati presentati molti nelle pagine delle varie riviste del settore, ma ben pochi presentano le caratteristiche di semplicità e praticità del progetto che stiamo presentando

Dovendo sostituire il radiocomando di una macchina industriale, macchinario ormai fuori produzione da tempo e ditta costruttrice fallita, si è resa necessaria una ricerca di mercato per tentare di sostituire il radiocomando ormai andato definitivamente. Con mia sorpresa mi sono accorto che prodotti industriali in grado di gestire otto o più canali ne esistono ben pochi. I modelli disponibili sono destinati al comando di gru o macchinari simili, utilizzati in campo edile. Oltretutto essendo destinati ad un mercato limitato e con caratteristiche di sicurezza molto elevate, hanno un costo decisamente proibitivo. È nata quindi la necessità di costruire qualcosa in casa per poter risolvere in maniera economica e definitiva il problema sorto.

I TRASMETTITORI

Scartata l'idea di utilizzare i classici integrati encoder e decoder dedicati all'utilizzo in questione, la soluzione più semplice ed economica resta quella del microcontrollore. L'idea di partenza relativa ad un progetto destinato a risolvere un determinato compito e morire con esso non mi era molto simpatica. È per questo che in fase di stesura del progetto, con la semplice aggiunta di un blocchetto di microinterruttori del modello usato dai radiocomandi classici e di un partitore resistivo recuperabile gratuitamente in qualsiasi matherboard o scheda video da buttare, è nato un progetto che può trovare un'infinità di applicazioni pratiche. Di trasmettitori ve ne presento due modelli, molto differenti tra loro ma con le stesse funzioni. Il primo modello, nonostante lo schema elettrico sia di una semplicità unica, richiede un maggiore impegno nella costruzione del contenitore dal punto di vista meccanico, pur essendo il più pratico da usare. Nel secondo sfruttiamo un contenitore commerciale della TEKO, il modello 10120/2, e uno schema elettrico appena più complicato che utilizza solo due pulsanti e un ponticello per la programmazione dell'indirizzo. Anche l'estetica ci guadagna, sembra un prodotto acquistato in negozio. Unica pecca l'utilizzo diventa più machiavellico. Per attivare un canale, bisogna prima selezionarlo con uno dei due pulsanti e quindi intervenire sull'altro per inviare la stringa di comando al ricevitore. L'ultimo canale selezionato resta comunque in memoria e richiamato alla pressione del pulsante di trasmissione; nel caso fosse necessario cambiare canale, la prassi resta comunque macchinosa. Probabilmente la mia opinione sarà sovvertita dal popolo di smanettoni di SMS che vivono in simbiosi con il loro cellulare e magari la seconda versione verrà apprezzata di più. Ve li presento entrambi, a voi la scelta.

SCHEMA ELETTRICO TX 1

Iniziamo dalla descrizione del primo trasmettitore. Per il modello costruito come sostituzione di quello guasto, ho utilizzato dei componenti che avrebbero garantito un'alta affidabilità del dispositivo e una certa velocità nella realizzazione pratica del trasmettitore. Mi riferisco all'utilizzo di una tastiera a membrana di tipo autoadesivo, componente molto pratico da utilizzare, esteticamente presentabile, ma decisamente costoso. Dopo aver avviato la macchina, che il capo reparto reclamava con urgenza tutti i giorni, ho deciso di riprogettare il trasmettitore utilizzando, questa volta, componenti molto semplici da reperire in commercio e dal costo abbordabile da qualunque hobbista. L'unico problema è una leggera complessità circuitale e un maggiore impegno nella foratura del contenitore. Non allarmatevi, la pulsanteria è l'unica parte complicata dell'intero progetto. Se date un'occhiata allo schema elettrico di **figura 1**, vi accorgerete che è di una banalità unica. Difatti la parte più delicata, ovvero la sezione che si occupa di inviare il segnale modulato nell'etere, è demandata ad un modulino molto economico. Con quattro saldature e uno spezzone di filo che funge da antenna, abbiamo risolto il problema RF. Il modulino in questione viene prodotto sia dalla Telecontrolli (modello RT5) che dalla Aurel (modelloTX-4MSIL), tra loro compatibili.

La parte importante del dispositivo è implementata da un PIC16F676, un microcontrollore molto conosciuto dai lettori.

Il micro adottato può essere sostituito dal modello PIC16F630, identico al PIC16F676 tranne per la mancanza del convertitore ADC, comunque non usato nel progetto. Ho utilizzato il modello completo unicamente per la più facile reperibilità sul mercato. Del resto, quasi tutte le periferiche contenute del micro non sono utilizzate.

Praticamente il PIC svolge tutte le funzioni richieste ad un trasmettitore per radio-

comando: testa i pulsanti e se ne trova uno premuto avvia un procedimento di preparazione del pacchetto da trasmettere e lo invia al modulo trasmettitore. Non solo, quando richiesto acquisisce un nuovo codice a 16 bit che diventa l'indirizzo del trasmettitore. Sfruttando l'area di memoria di tipo EEPROM, salva questo codice in modo stabile, così anche il cambio batteria non lo può cancellare.
Trattandosi di un dispositivo alimentato a batteria, diventa di primaria importanza minimizzare il più possibile il consumo di energia. A tal fine viene utilizzata una funzione presente in tutti i micro della Microchip, ovvero l'istruzione SLEEP. Quando il PIC incontra quest'istruzione, disattiva l'oscillatore e si pone in basso consumo. L'assorbimento di energia del micro in SLEEP è talmente basso da non riuscire a misurarlo con un comune tester. Ovviamente per poter leggere la tastiera nella situazione di Power-down, è necessario posizionare i tre piedini 8, 9 e 10, che nella matrice formano le righe dei pulsanti, a livello logico alto. Attivando l'interrupt associato al cambio di fronte sulla PORTA A, possiamo risvegliare il PIC in qualsiasi momento premendo qualsiasi pulsante. Quando si usa la funzione SLEEP, è importante ricordarsi di parametrizzare come ingresso i piedini del PIC non usati e collegarli al positivo o a massa. Se non usate questo accorgimento, vi ritroverete con il micro che as-

LISTA COMPONENTI Tx 1

R1-2	330 Ω 1/4 W
R3-5	22 KΩ 1/4 W (1/8W)
C1	47 µF 16 V elettrolitico
C2	100 nF poliestere o ceramico
D1	LED rosso 3 mm
D2	LED verde 3 mm
U1	PIC 16F676
M1	Modulo trasmettitore 433 MHz (RT5 Telecontrolli)
P1-9	Pulsante da CS L=7mm
Contenitore SC700 (TEKO 10015) (mis. 90x56x22)	

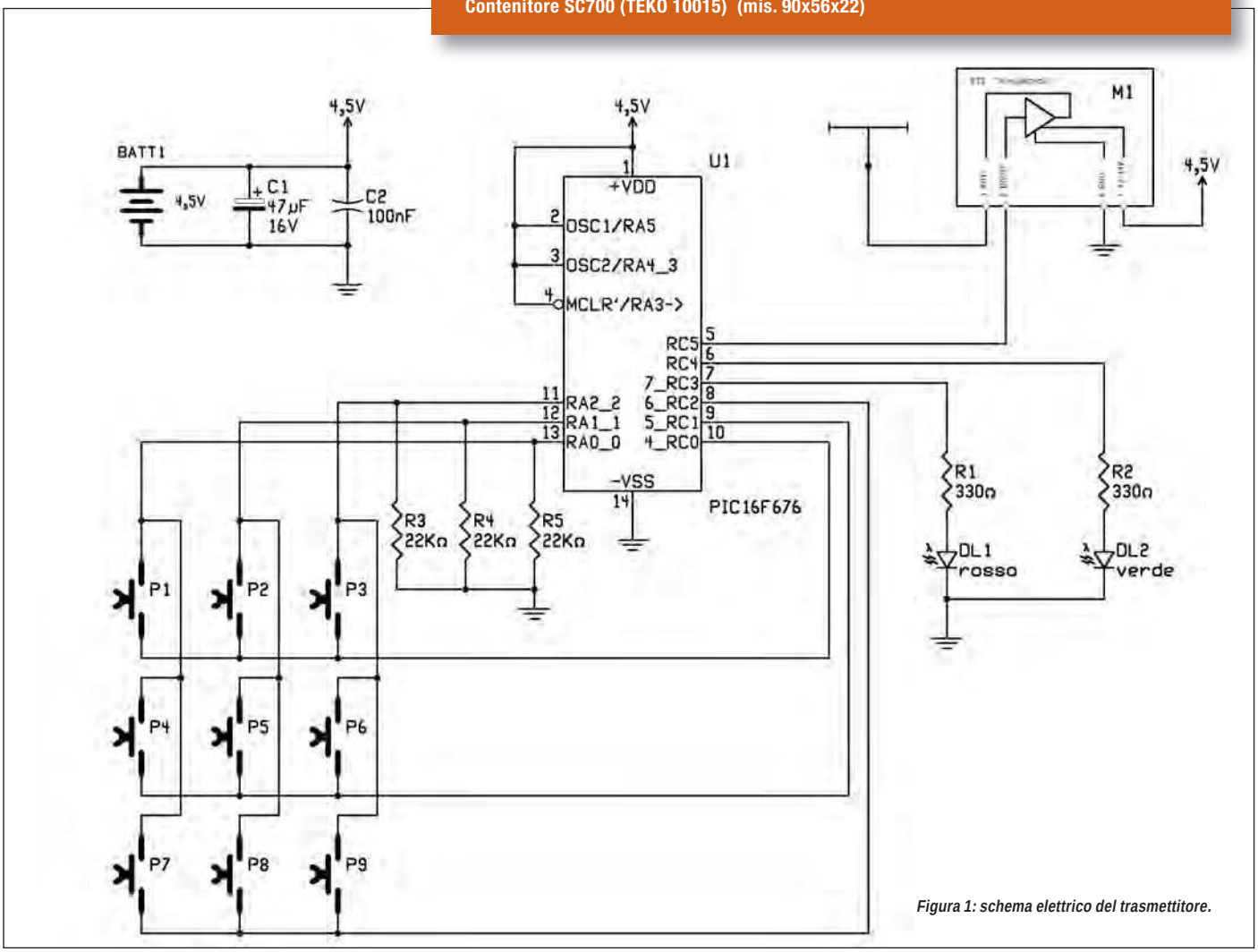


Figura 1: schema elettrico del trasmettitore.

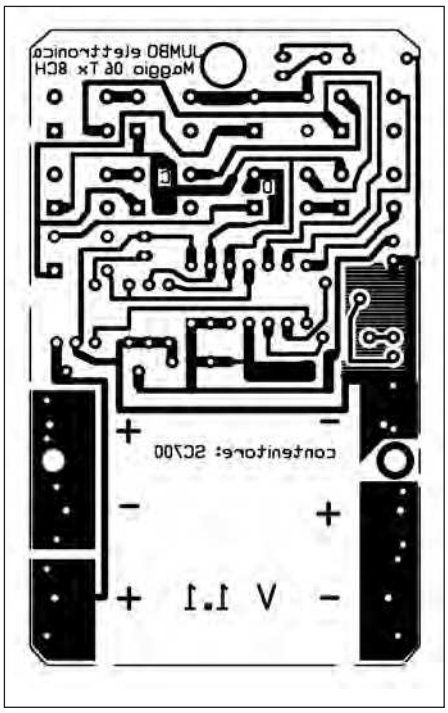


Figura 2: circuito stampato del trasmettitore.

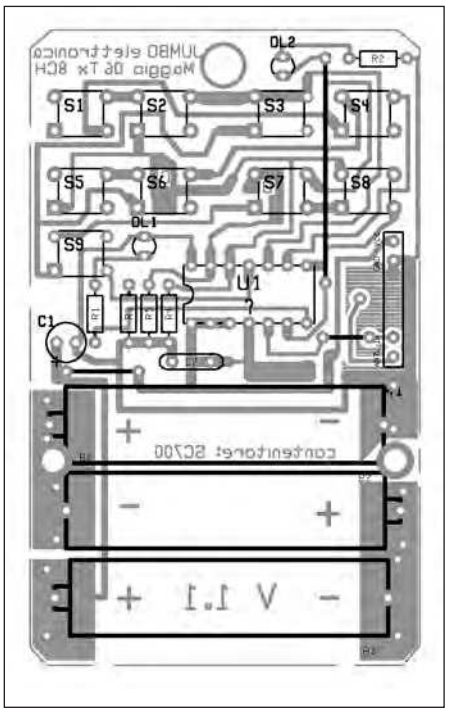


Figura 3: piano di montaggio dei componenti per il circuito di figura 1.

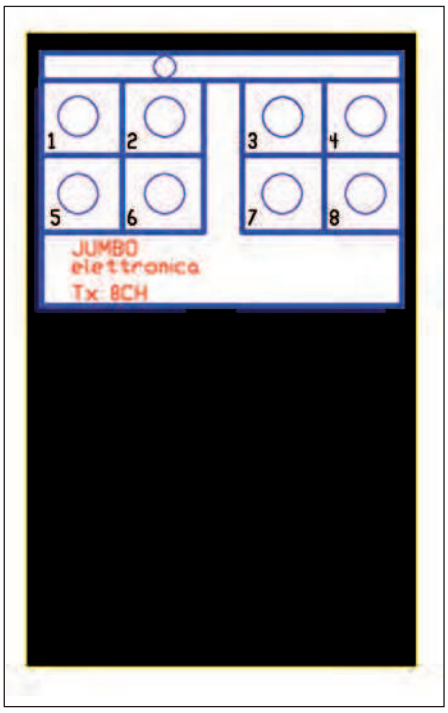


Figura 4: la maschera per il telecomando.

sorbe una corrente variabile nel tempo, anche se di pochi microamper.
L'alimentazione è garantita da tre pile formato mini stilo che assicurano una lunghissima autonomia (diversi anni) con un utilizzo normale.

PROTOCOLLO SOFTWARE

Il protocollo adottato è composto da 42 bit trasmessi seguendo il seguente andamento temporale:
1 bit start
1 bit repeat
16 bit indirizzo
8 bit canale (pulsante 1 = OUT 1 = bit 0; ...; pulsante 8 = OUT 8 = bit 7)
8 bit codice variabile
8 bit indice codice variabile
Vediamo in dettaglio ogni singola voce del protocollo.
Il bit di start serve unicamente ad indicare al ricevitore che un pacchetto sta per essere inviato.
Il secondo bit, quello di repeat, è a 0 quando viene inviato il pacchetto per la prima volta, nelle volte successive questo bit viene posto a 1. Si intende invio continuo e ripetuto dei pacchetti quando viene mantenuto premuto lo stesso pulsante. Nel progetto tale funzione non viene utilizzata, ma è stata implementata unicamente per il comando in sicurezza nell'ambiente industriale.
I 16 bit dell'indirizzo sono personalizzabili attraverso una procedura molto semplice che vedremo più avanti, con l'unico utilizzo dei pulsanti del trasmettitore. Sono salvati direttamente nella EEPROM del PIC e memorizzati nel ricevitore in fase di acquisizione.
Gli 8 bit contengono lo stato dei pulsanti del trasmettitore. Ad ogni bit è assegnato un pulsante, il bit 0 corrisponde al pulsante 1, il bit 1 al pulsante 2 e così via. Nel ricevitore le otto uscite sono assegnate agli 8 pulsante del trasmettitore con la stessa numerazione.
Gli 8 bit del codice variabile e gli 8 bit dell'indice sono stati inseriti per aumentare la sicurezza dell'accoppiata trasmettitore/ricevitore. In pratica il trasmettitore, sfruttando il TMR0, genera in maniera casuale un numero da 1 a 64 che viene chiamato *indice codice variabile*. Tale indice va a leggere una tabella che contiene 64 *codici variabili* diversi. In fase di trasmissione sia l'indice che il codice

CODICE MIP 2797066

corrispondente trovano posto nella stringa trasmessa e il compito del ricevitore è quello di confrontare il codice variabile ricevuto con quello presente nella sua tabella, identica a quella del trasmettitore, utilizzando come indice proprio quello trasmesso nel protocollo.
Il tempo medio per la trasmissione di una stringa completa è di circa 42 ms.
Per comodità, la parte di codice che contiene la ricezione del pacchetto è stata inserita in un file a parte con il nome di "*Rx.inc*", mentre quella del trasmettitore si trova alla subroutine "*_Tx*". In questo modo risulta utile e comodo riutilizzare le routine anche in programmi personali, e risulta più facile modificare il protocollo.

PROTOCOLLO HARDWARE

Inizialmente avevo pensato di utilizzare una semplice trasmissione seriale del tipo usato dal RS232 del PC, ma ricerche relative ad un precedente progetto mi avevano portato alla conoscenza della codifica di tipo KEELOQ della Microchip.
Il protocollo hardware utilizzato è molto intelligente perché permette la sincronizzazione non più unicamente dal bit di start ma su ogni singolo bit. Questo vuol dire che non è più necessario contare sulla precisione del clock del micro, ma è sufficiente che trasmettitore e ricevitore abbiano una frequenza di clock simile.
Ogni singolo bit nel sistema KEELOQ è diviso in tre parti di 400 microsecondi ciascuno **(figura 5)**, il primo sempre alto (1 logico), la parte centrale è posto a livello alto se il bit vale 0, oppure a 0 se il bit vale 1, la terza parte sempre a 0 in modo da preparare il sincronismo al bit successivo.
Il protocollo adottato nel progetto è leggermente modificato dallo standard KEELOQ, permettendo un leggero aumento della velocità di trasmissione **(figura 6)**.
Anche nel nostro caso, la prima parte del bit è sempre alta (1 logico), la seconda è alta se viene trasmesso un 1 logico, mentre è bassa se viene trasmesso lo 0 logico. La terza parte esiste solo nel caso del 1 logico ed è posta a 0, mentre nel caso il bit valga 0, non esiste la terza parte. In questo modo viene garantito il sincronismo su ogni singolo bit, diminuendo il tempo di trasmissione/ricezione del pacchetto.

MONTAGGIO TX 1

Il montaggio del circuito non presenta particolari difficoltà, ma vi consiglio comunque la costruzione del circuito stampato proposto in **figura 2**, peraltro scaricabile gratuitamente dal sito della rivista. Inoltre è previsto il montaggio in un contenitore estremamente economico.
Una volta in possesso del circuito stampato, montare i pochi componenti richiede pochissimo tempo. Se volete, potete utilizzare lo zoccolo per il microcontrollore. Particolare cura deve essere posta nel montaggio dei pulsantini. I primi 8 vanno montati dal lato rame, mentre il nono, quello necessario all'inserimento dell'indirizzo, dal lato componenti. I pulsanti indicati presentano un'altezza totale, piolino e corpo, di 7 mm, quanto basta a sfiorare la superficie esterna del contenitore una volta che il tutto sarà assemblato. Anche la saldatura dei pulsanti da 1 a 8 va eseguita con molta cura dal lato rame. Raccomando la saldatura di tutti i pins di ogni singolo pulsante, dato che viene sfruttato il loro collegamento interno: se ve ne dimenticate qualcuna, il circuito non può funzionare. Le **figure 7**,**8** e **9** chiariscono ogni dubbio.
Per quanto riguarda le batterie, dalle prove effettuate e da esperienze passate, possono essere semplicemente saldate direttamente allo stampato con dei piccoli spezzoncini di rame nudo, per esempio quello eccedente dalle resistenze. L'uni-

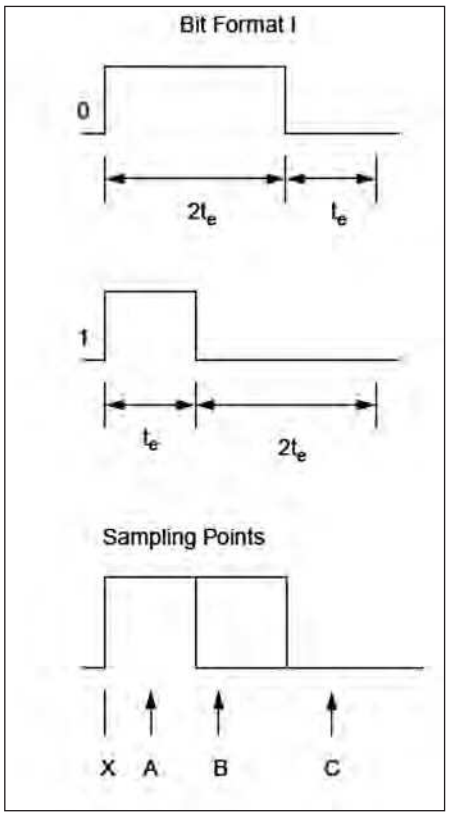


Figura 5: timing del protocollo Keeloq.

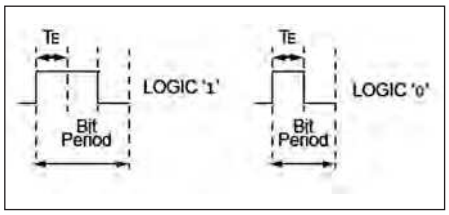


figura 6: il protocollo Keeloq modificato per il progetto.

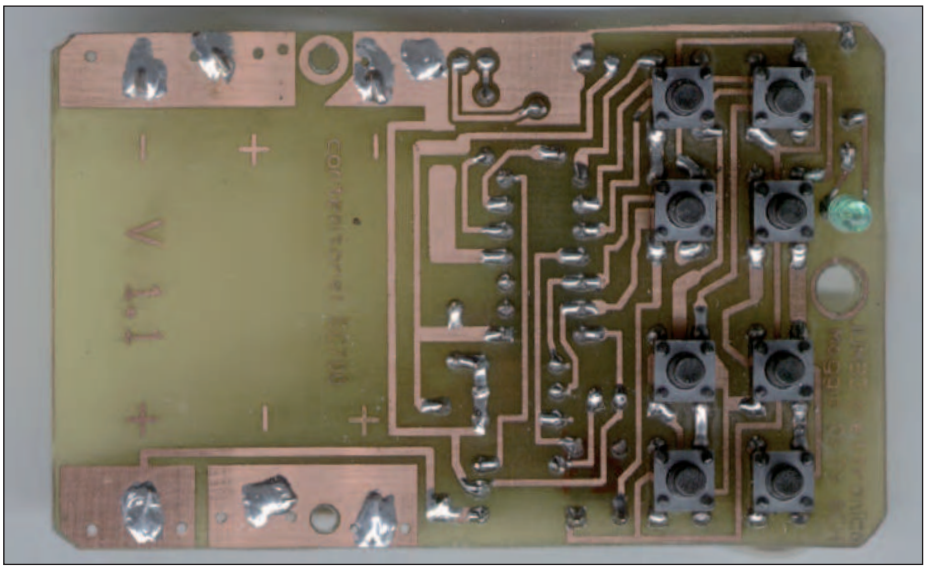


Figura 7: il posizionamento dei pulsanti.

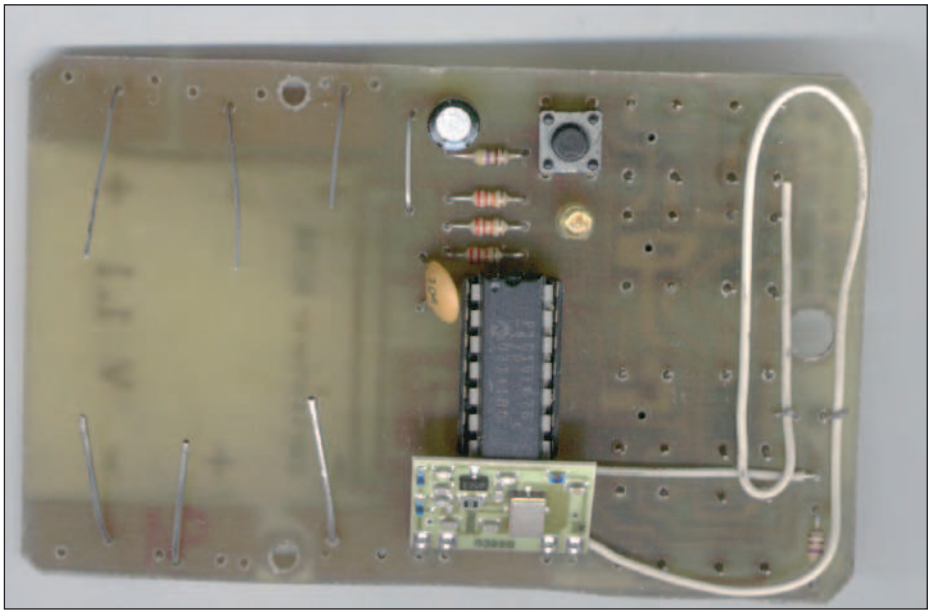


Figura 8: il circuito a montaggio ultimato.

Figura 9: il circuito stampato inserito nella scatola.

ca accortezza necessaria è di non riscaldare eccessivamente le batterie durante la saldatura.
Nonostante la soluzione vi sembri un po' rozza e poco pratica nel caso si renda necessaria la sostituzione, vi posso assicurare che con un utilizzo casalingo queste avranno una durata di diversi anni. Personalmente, nei miei prototipi ho usato le pile smesse dagli MP3 dei miei figli e continuano a funzionare regolarmente da più di un anno. Non dimenticate di saldare uno spezzone di filo lungo 17 cm che funge da antenna sulla piazzola che trovate tra il PIC e il modulino M1 (uscita *antenna* di M1). Ripiegatela su se stessa appoggiandola allo stampato all'altezza della prima fila di pulsanti. Questo accorgimento migliorerà le prestazioni riguardanti la distanza operativa.
È stato previsto l'alloggiamento in un contenitore di tipo SC-700 di colore nero e dalle dimissioni di 90 x 56 x 22. La scelta di tale contenitore è dettata da una diffusione pressoché totale dell'articolo, si trova in qualsiasi negozio di elettronica e in tutte le fiere del settore e, fatto positivo, dal costo contenuto.
Come prima operazione vi consiglio di stampare in doppia copia la mascherina proposta in **figura 4**, eventualmente dopo che ognuno avrà avuto modo di personalizzarla secondo le proprie esigenze o gusti. Come supporto potete usare delle etichette autoadesive di grandi dimensioni oppure, se volete ottenere un risultato esteticamente migliore e duraturo, della carta adesiva per uso fotografico. Quest'ultima si trova in tutti i centri commerciali ad un costo non propriamente economico, ma il risultato finale è decisamente migliore.
La prima copia autoadesiva della mascherina va attaccata con cura al coperchio del contenitore, vi servirà da dima per la foratura. Per il foro del LED è sufficiente una punta da 3 mm, mentre per i pulsanti è necessario un foro dal diametro di 6-7 mm. A questo punto sbavate i fori con una punta più grossa, 10 mm va benissimo, facendo molta attenzione perché la plastica del contenitore è molto morbida.
L'operazione di sbavatura va fatta a mano e senza premere molto la punta sulla plastica, altrimenti correte il rischio di trovarvi con un foro ben più grosso di quello prospettato. Sostituite l'adesivo ormai rovinato dal trapano con il secondo, avendo l'accortezza di forarlo solo dove uscirà il LED; i pulsanti verranno azionati da sopra l'adesivo, come avviene nei modelli commerciali.
Montate il tutto e, se non avete ancora fatto, inserite il PIC nel proprio zoccolo, dopo averlo programmato con il file "*8CH_Tx.HEX*" che potete scaricare dal sito della rivista.
Controllate che la parte superiore dei pulsanti sfiori la parte adesiva del coperchio senza toccarla o in maniera lieve. Se tutto è a posto, premendo qualsiasi pulsante da 1 a 8 deve lampeggiare il LED verde, mentre il nono pulsante (SET), accessibile solo dall'interno del contenitore, accende il LED rosso. Due gocce di colla a caldo sono necessarie per bloccare il CS al contenitore e permettere a tutti i pulsanti di essere premuti.

PROGRAMMAZIONE TX 1

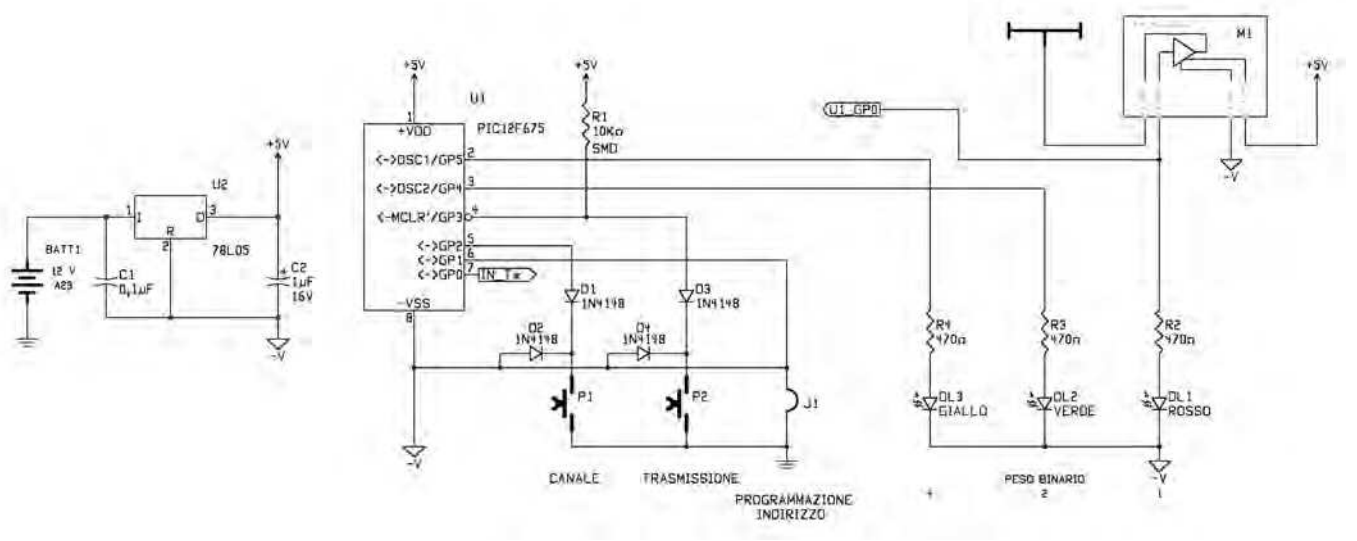
Come accennato nei capitoli precedenti, nel protocollo sono presenti 16 bit che rappresentano l'indirizzo univoco di ciascun trasmettitore.
Avrete certamente notato la mancanza del dip switch, generalmente presente nei radiocomandi, necessario all'impostazione del codice. Nella nostra applicazione utilizziamo semplicemente il pul-

sante P1 (canale 1) per impostare lo 0 logico, e il pulsante P4 (canale 4) per 1 logico. La stringa di 16 bit da inserire nella radiocomando viene letta da sinistra verso destra, cioè il primo pulsante premuto programma il 16° bit della stringa, mentre alla sedicesima pressione viene programmato il bit 0. Per esempio se desiderate inserire il seguente indirizzo
[1110100100011011]
leggendo la stringa da sinistra verso destra dobbiamo premere la seguente sequenza di pulsanti
[P4-P4-P4-P1-P4-P1-P1-P4-P1-P1-P1-P4-P4-P1-P4-P4]
naturalmente durante la fase di programmazione.
Per accedere alla fase di programmazione dell'indirizzo, è sufficiente premere il pulsante P9 (SET) e tenerlo premuto finché il LED rosso passi da una condizione di luce fissa a lampeggiante (circa 3 sec). Avete a disposizione una decina di secondi per premere uno dei due pulsanti riservati alla programmazione. Permette, anche a caso, il pulsante del canale 1 e quello del canale 4 nella maniera che più vi aggrada, per un totale di 16 azionamenti. Lo spegnimento del LED rosso, indica la fine della fase di programmazione. Il codice inserito viene memorizzato nella EEPROM del PIC in maniera permanente. Nel caso lasciaste passare più di dieci secondi tra la pressione di un pulsante e l'altro, il microcontrollore riconosce la situazione di time-out ed esce dalla fase di programmazione, conserva il vecchio indirizzo e si pone nella condizione di SLEEP.

SCHEMA ELETTRICO TX 2

Proseguiamo in maniera spedita alla descrizione del secondo trasmettitore.
A questo punto possiamo omettere la questione protocollo perché identico alla versione precedentemente descritta. Anche il modulino RF è lo stesso, mentre il PIC, pur restando nella stessa famiglia, è un modello a otto piedini. Il PIC 12F675 non è altro che la versione usata nel Tx 1 con un numero di pins ridotto ma sufficiente alla gestione del trasmettitore. Anche le periferiche sono le stesse.
Dallo schema elettrico di **figura 10** notiamo la presenza di due diodi (vedi D2 e D4), con il compito di alimentare l'intero circuito alla pressione di uno dei due pulsanti, mentre la presenza del ponticello J1 assicura l'alimentazione durante l'intera fase di programmazione dell'indirizzo. Si è resa necessaria una soluzione di questo tipo a causa del tipo di alimentazione, una micro pila da 12V che il contenitore ci obbliga ad usare. Considerata la presenza di un regolatore di tensione che deve ridurre da 12 a 5 Volt l'alimentazione e consumando circa 3 mA il solo regolatore (uno zener richiede più corrente), la micro pila sarebbe durata pochissimo. Con la soluzione adottata, l'intero trasmettitore viene regolarmente alimentato solo premendo uno dei due pulsanti o inserendo il ponticello J1.
I due diodi D1 e D3 hanno lo scopo di fare capire al PIC quale dei due pulsanti è stato premuto.
Come descritto all'inizio dell'articolo, il pulsante P1 ha lo scopo di selezionare il canale da attivare (o disattivare); per la verifica ci serviamo di tre LED come monitor di selezione. Il sistema usato per ridurre al minimo il numero dei componenti impiegati è quello del peso binario. In pratica ogni LED assume un valore (peso) e la

Figura 10: schema elettrico alternativo per il trasmettitore.



LISTA COMPONENTI Tx 2	
R1	47KΩ SMD (vedi testo)
R2-4	180 Ω 1/4 W (1/8W)
C1	100 nF multistrato
C2	1 µF 16 V elettrolitico (vedi testo)
DL1	LED rosso 3 mm
DL2	LED verde 3 mm
DL3	LED giallo 3 mm
U1	PIC 12F675
U2	78L05
M1	Modulo trasmettitore 433 MHz (RT5 Telecontrolli)
P1-2	Pulsante da CS
Contenitore TEKO 10120/2 (mis. 61x37x15)	

somma dei LED accesi determina il numero del canale selezionato. In particolare il LED rosso (DL1) vale 1, mentre il verde (DL2) vale 2 e quello giallo (DL3) vale 4. Quando tutti i LED sono spenti il valore totale sarà 0, il che significa canale 1 selezionato, con tutti i LED accesi il valore binario sarà 7, cioè canale 8 selezionato. Facciamo un esempio pratico: premiamo P1 fino ad accendere il LED DL2 e DL3. DL2 vale 2 mentre DL3 vale 4, la somma è 6 e il canale coinvolto sarà il 7 (attenzione nella numerazione binaria lo 0 è significativo).

Ad ogni pressione del P1, il valore binario viene incrementato di una unità fino ad accendere tutti i LED indicando il valore binario 7 (canale 8). Alla pressione successiva di P1 il contatore interno al PIC ritorna a 0 e i LED restano spenti, e così via in un loop continuo.

Il pulsante P2 semplicemente invia la stringa al ricevitore ripetendola ogni 100 ms. Da notare che anche in questo caso i LED visualizzano il canale selezionato con il metodo appena visto, l'unica differenza è il LED rosso (DL1) che lampeggia con la cadenza della stringa trasmessa. È un "difetto" dovuto al numero limitato di piedini del PIC.

Terminiamo l'analisi dello schema elettrico con il regolatore U2 il cui compito è quello di ridurre la tensione da 12 V della micro pila a 5 V, ideale per alimentare correttamente il PIC e il modulino trasmettitore. Importante è il collegamento del terminale 2 che generalmente va collegato direttamente a massa: in questo caso il collegamento a massa avviene solo in presenza del ponticello J1 o in conseguenza della pressione di P1 o P2.

MONTAGGIO TX 2

La presenza di un contenitore commerciale richiede la preparazione dello stampato con la disposizione dei componenti piuttosto rigida. I pulsanti in particolare devono trovarsi in una ben determinata posizione, altrimenti non verranno azionati. Una zona dello stampato, appena sotto i LED, va lasciata libera permettendo l'appoggio del supporto mobile che andrà ad azionare i pulsanti. Tenete presente che alcuni componenti come il condensatore C2 possono avere dimensioni tali da impedire la chiusura del coperchio. Per questi motivi vi invito alla costruzione dello stampato sul disegno proposto in **figura 11**.

Una volta in possesso del CS inciso e forato, vi consiglio di iniziare il montaggio dai diodi e resistenze. Con i terminali in eccesso delle resistenze, costruite i tre ponticelli e saldateli nella posizione indicata dal piano pratico di **figura 12**. Uno di questi, che va posto proprio sotto il P2, va assolutamente fatto prima di saldare il pulsante. Proseguite con i pulsanti, il ponticello J1, i condensatori e il regolatore. Quest'ultimo va adagiato sulla superfi-

Figura 11: circuito stampato per lo schema elettrico di figura 10.

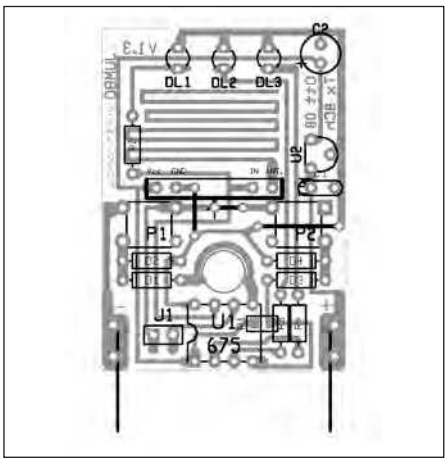


Figura 12: piano di montaggio dei componenti.

cie del CS prima della saldatura, regolatevi come da **figura 14**. Il C2 se non lo trovate di piccole dimensioni (personalmente ho usato un modello al tantaglio), potete sostituirlo con un modello ceramico o multistrato da 100 nF. Lo zoccolo del micro deve essere a basso profilo. Nel caso non lo troviate, il micro va saldato direttamente allo stampato dopo averlo programmato. Il programma da caricare in questa versione è "*Tx_8canali_V1_3.hex*". Al contrario del modello precedente e del ricevitore, questa versione è programmata con il MikroC di Mikroelektronika.

Essendo il programma di poco superiore ai 500 Byte, potete tranquillamente modificarlo e compilarlo anche con la versione fornita gratuitamente.

Il modulo trasmettitore va montato dal lato rame con i componenti in bella vista **(figura 13)**. I terminali vanno ripiegati

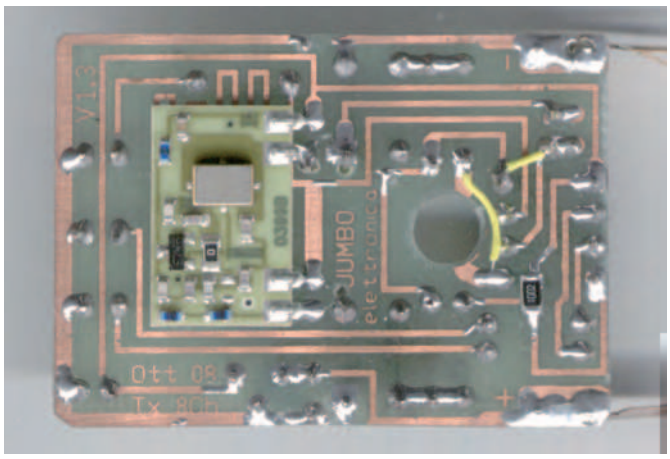


Figura 13: il modulo trasmettitore va montato dal lato rame.

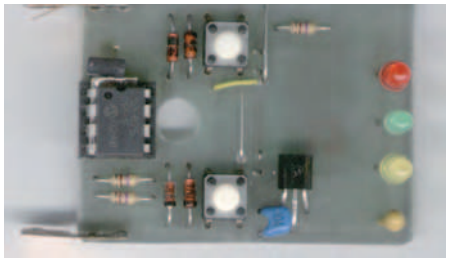


Figura 14: dettagli del montaggio della scheda.

molto corti e la parte piana del modulo deve sfiorare il rame del CS. Purtroppo questa è l'unica posizione per farci stare il modulo RF all'interno del contenitore proposto. Dal lato rame trova posto anche la resistenza R1 versione SMD. Non sono riuscito a fare diversamente se non disegnando un CS doppia faccia,e la cosa non mi era simpatica. Comunque il valore non è impegnativo, da 10 a 47 K sono valori accettabili. L'unico scopo di tale componente è la forzatura al positivo del piedino 4 (GPIO 3), unico ingresso del PIC al quale non sia collegata internamente la resistenza di Pull-up.

Le due pagliuzze di rame che formano i contatti per la batteria vengono fornite a corredo del contenitore.

PROGRAMMAZIONE E USO TX 2

La fase di programmazione risulta essere molto semplice. Dopo avere inserito il jumper J1, il LED verde (DL2) lampeggia con cadenza di un secondo confermando la fase di programmazione. Premendo P1 si accende il LED rosso (LD1) per tutto il tempo in cui resta premuto il pulsante e viene assegnato uno "0", mentre premendo P2 si accende il LED giallo (DL3) e viene assegnato un "1". Per primo viene programmato il bit 15 (MSB), per ultimo il bit 0 (LSB). Una volta programmati tutti i 16 bit, l'indirizzo viene salvato nella memoria EEPROM del PIC e il LED verde lampeggia velocemente indicando la fine della fase di programmazione. Tolto il ponticello, il nostro trasmettitore è pronto al funzionamento. Il pulsante P2 ha il compito di inviare la stringa contenente il canale da attivare/disattivare, che è sempre l'ultimo selezionato da P1 e salvato in memoria. Nel caso sia necessario gestire un canale diverso da quello in memoria, prima deve essere selezionato da P1 e visualizzato dai tre LED presenti nel trasmettitore con la modalità descritta nella parte riguardante lo schema elettrico. Al fine di dissipare ogni dubbio, procediamo con un esempio pratico. Immaginiamo di dovere controllare il canale 3 del ricevitore, premendo P1, incrementa in sequenza il valore binario visualizzato dai tre LED. Ci dobbiamo fermare quando resta acceso solo il LED verde (DL2) il quale indica la selezione del canale 3 e il relativo salvataggio in memoria. Premendo P2, la stringa viene inviata al ricevitore con il compito di controllare il canale 3. Se non viene effettuata una nuova selezione, ogni volta che premiamo P2, siamo in grado di controllare unicamente lo stesso canale. Non mancate di leggere l'articolo riguardante il ricevitore perché offre molti spunti per le vostre applicazioni future, come la possibilità di utilizzare la stessa porta del micro come uscita pilotando dei relè e nello stesso tempo come ingresso testando lo stato di un dip switch. ❑

CODICE MIP 2804835

Sensori per TUTTI

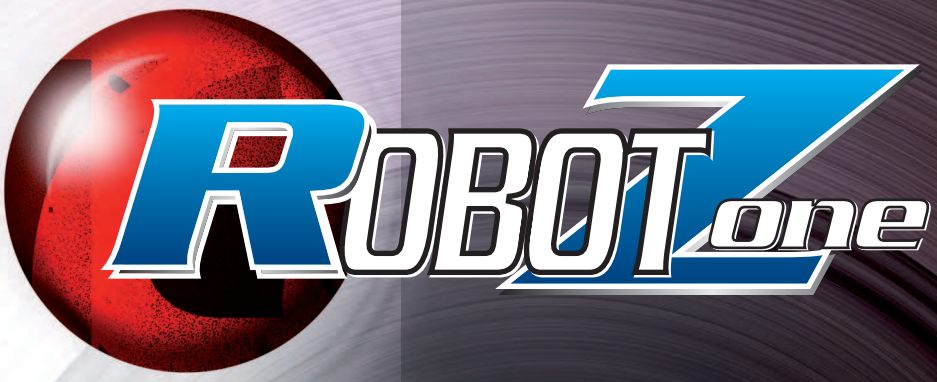
Se il tuo problema è aggiungere e testare rapidamente un sensore con il tuo prototipo, forse non sapevi che Parallax ne fornisce di convenienti e pronti all'uso: accelerometri, temperatura, umidità, GPS e posizionamento, rotazione, distanza, colore e luce, pressione, movimento, gas, suono, e molti altri.

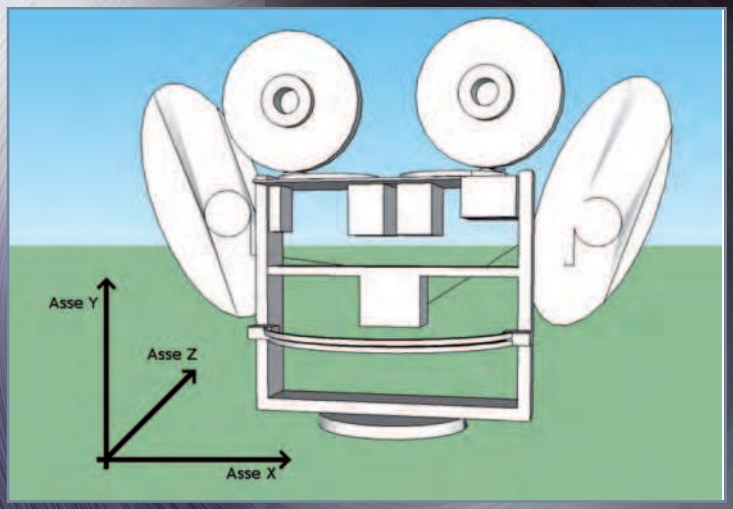
Distribuito da **elettroshop.com**
tel. 02 66504794 - fax 02 66508225
www.elettroshop.com/sensori

PARALLAX INC
www.parallax.com
Friendly microcontrollers, legendary resources.™

Parallax ed il logo Parallax sono marchi registrati di Parallax Inc.

CODICE MIP 2808469





102 ROBOT GOLEM (parte quarta)

GOLEM: LA TESTA

In questo articolo vedremo in dettaglio la costruzione della testa del robot, installandovi le webcam, i microfoni e l'altoparlante

di Mattias Costantini

Robot Golem *(parte quarta)*

GOLEM: la testa

In questo articolo vedremo in dettaglio la costruzione della testa del robot, installandovi le webcam, i microfoni e l'altoparlante

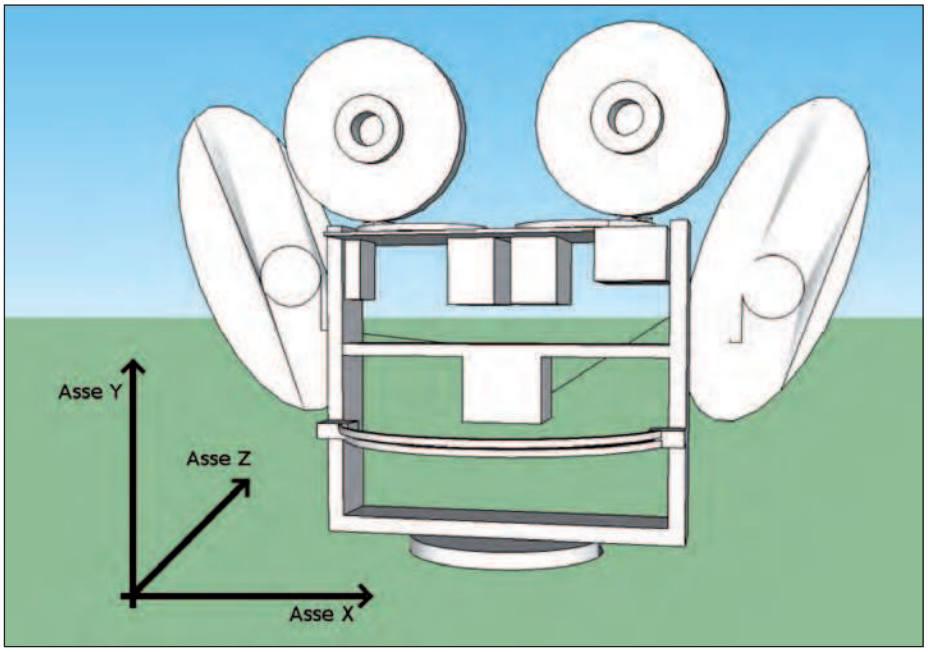


Figura 1: immagine 3D della testa con relativa posizione rispetto agli assi X, Y e Z.

Siamo arrivati al quarto articolo dedicato al Robot Golem. Nel primo articolo ho dato un'ampia panoramica del progetto, descrivendone gli obiettivi, spiegando cosa può fare e perché sia stato rilasciato sotto licenza open source.
Nel secondo articolo abbiamo iniziato ad avventuraci nella costruzione del robot: il Golem è un robot modulare, sia dal punto di vista meccanico che dal punto hardware/software. Esso è composto da una serie di moduli che possono essere assemblati l'uno con l'altro, come delle costruzioni. I vari moduli hanno in comune, oltre alla forma estetica, anche i bus di comunicazione che nel caso specifico sono due: il bus di alimentazione e il bus dati. Il bus di alimentazione svolge due funzioni principali, portando:

Figura 2: immagine di un servomotore da modellismo con una squadretta in alluminio

- l'alimentazione necessaria al corretto funzionamento di attuatori come motori elettrici;
- la tensione necessaria alle viarie logiche di controllo del sistema.

Il bus di comunicazione, invece, serve per poter interfacciare e mettere in comunicazione tutti i moduli del Golem attraverso un apposito bus, chiamato Gbus, basato sullo standard multicast CAN (Controller Area Network). Consiglio ai lettori che non hanno letto i precedenti articoli di reperire tutti quelli mancanti, in questo modo sarà più facile seguire il filo logico di questo articolo.

INTRODUZIONE

Nello scorso articolo, ho accennato al forte impulso evolutivo che ha attraversato la robotica domestica nel corso degli ultimi anni. I robot non assomigliano più a dei "barattoli di latta" come nei film degli anni '60, ma loro fisionomia si sta sempre di più adattando ai nostri canoni estetici. Questo è necessario non solo dal punto di vista visivo, ma, soprattutto, da un punto di vista interattivo: siamo molto più propensi ad utilizzare una forma che già conosciamo piuttosto che stringere amicizia con un soggetto "alieno".
Questo è uno dei motivi principali che mi ha spinto a dotare il Golem di una testa vera e propria, che riesca ad incuriosire ed attrarre le persone. La testa che ho realizzato è alla portata di tutti: non ho utilizzato materiali o componenti insoliti, mi sono limitato a quelli disponibili a tutti, se-

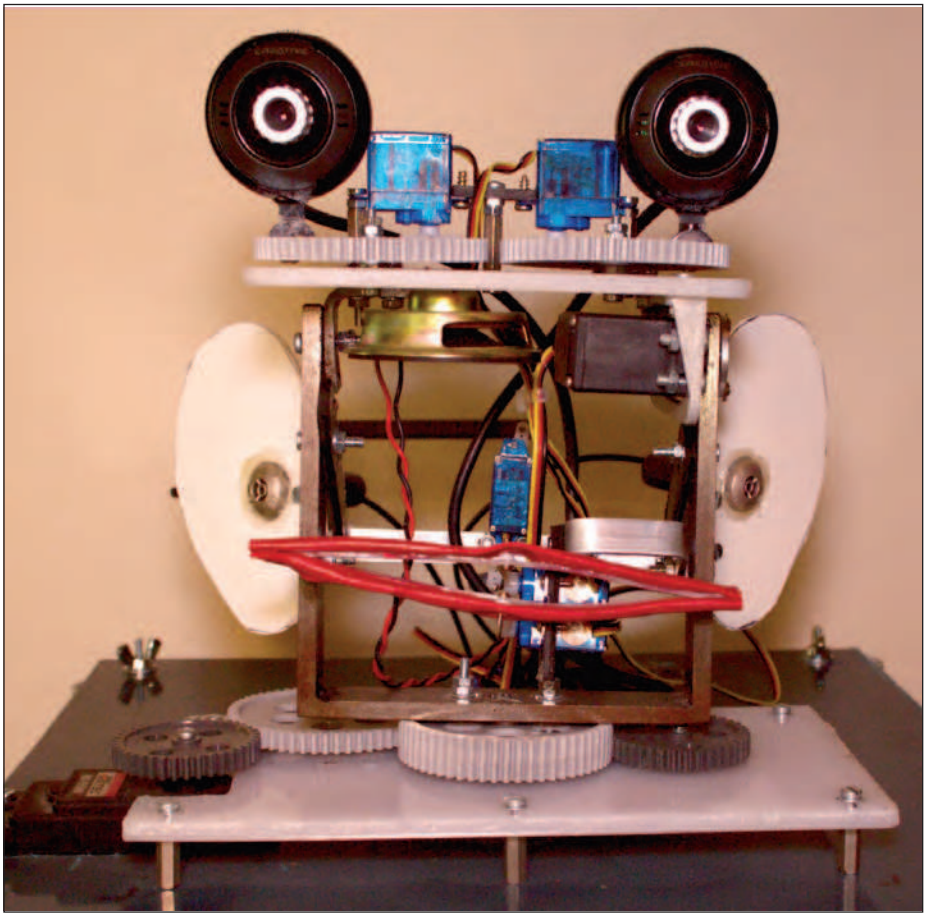


Figura 3: vista anteriore della testa del robot.

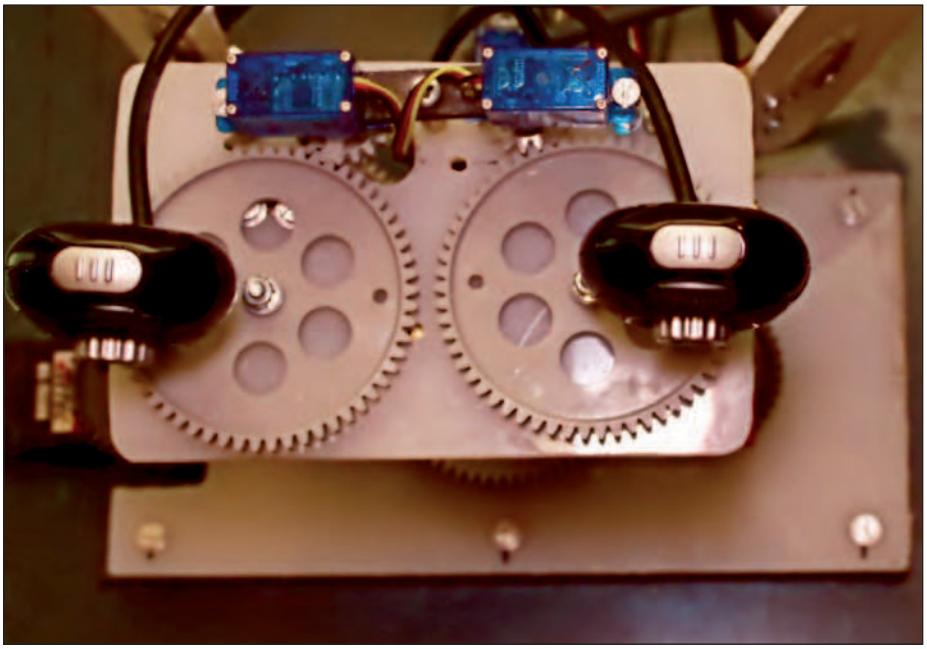


Figura 4: vista dall'alto della testa del robot, si notino gli ingranaggi per la convergenza/divergenza delle webcam, mossi da due micro servi.

guendo la filosofia open source, ovvero la sua costruzione deve essere alla portata del maggior numero di persone possibili. Ecco che webcam, PVC, plexiglass, fili, dadi, viti, ingranaggi e servomotori da modellismo si uniscono creando un sistema dinamico, aggraziato e semplice. Tutti i "gradi di movimento" (DoF, degree of freedom) sono stati realizzati grazie a dei servomotori montati su apposite squadrette o su ingranaggi.

CHE COSA SONO I SERVOMOTORI?

Un servomotore è un attuatore. Generalmente, si presenta come un contenitore di materiale plastico dal quale esce un cavo di alimentazione e di controllo, assieme ad un pignone dentellato.
All'interno dell'involucro, trovano posto una circuiteria elettronica dedicata, un potenziometro, un motore elettrico alimentato con una tensione continua di 5/6 volts ed una serie di ingranaggi. Il principio di funzionamento è molto semplice: grazie ad un apposito segnale, il pignone esterno si sposta in un angolo compreso tra -90° e 90°.
Il potenziometro è collegato meccanicamente al motore e serve per misurare l'attuale angolo di spostamento, se all'impulso inviato al servomotore corrisponde un angolo diverso di spostamento, esso ruoterà sino al raggiungimento dell'angolo di lavoro impostato.
Per permettere il movimento, oltre a fornire le giuste tensioni di alimentazione, viene dato come input un treno di impulsi, composto da un'onda quadra compresa in un range tra 600usec e 2100 usec, e tra i vari impulsi la pausa deve essere di circa 20 ms.
L'intervallo di tempo variabile necessario al movimento del servomotore lo chiamerò At.

COME CALCOLARE VELOCEMENTE At

Quando sappiamo quanto deve valere in termini di tempo At, dobbiamo calcolarlo in base alla nostra configurazione hardware.
La scheda Freedom I del Golem monta un PIC Microchip 18F4580 con un clock settato a 20 MHz, tramite il quale si deve

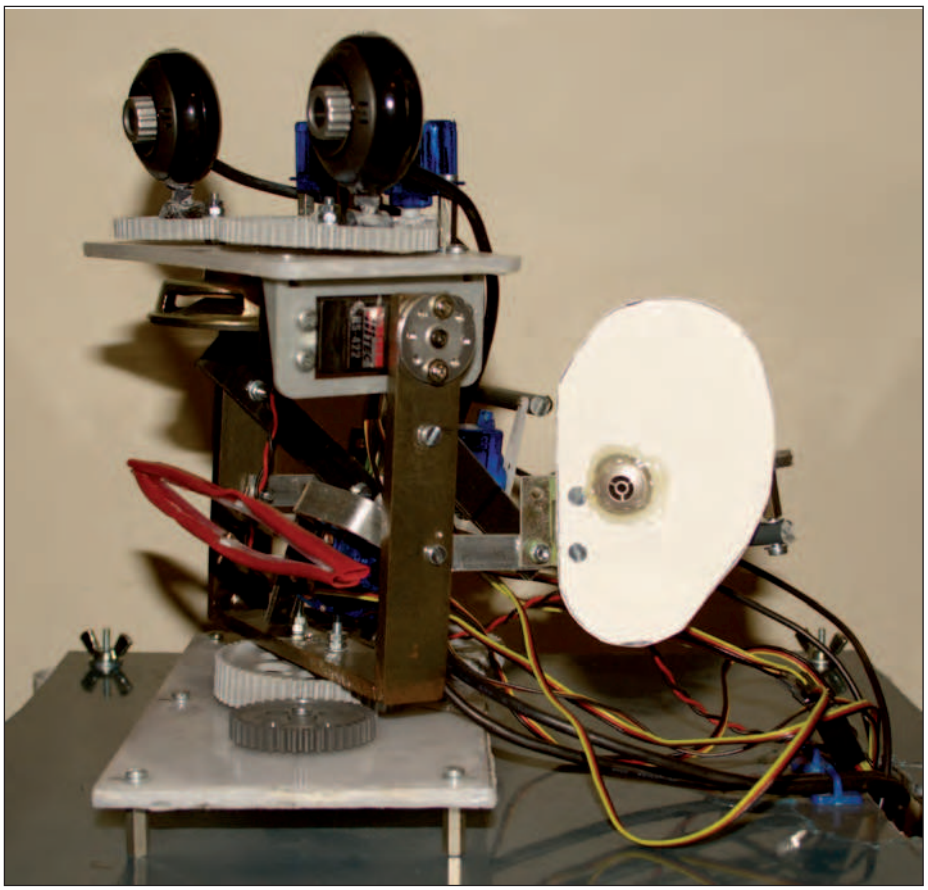


Figura 5: vista di lato della testa del robot, la posizione del microfono rispetto all'orecchio rispecchia la proporzione del condotto uditivo umano.

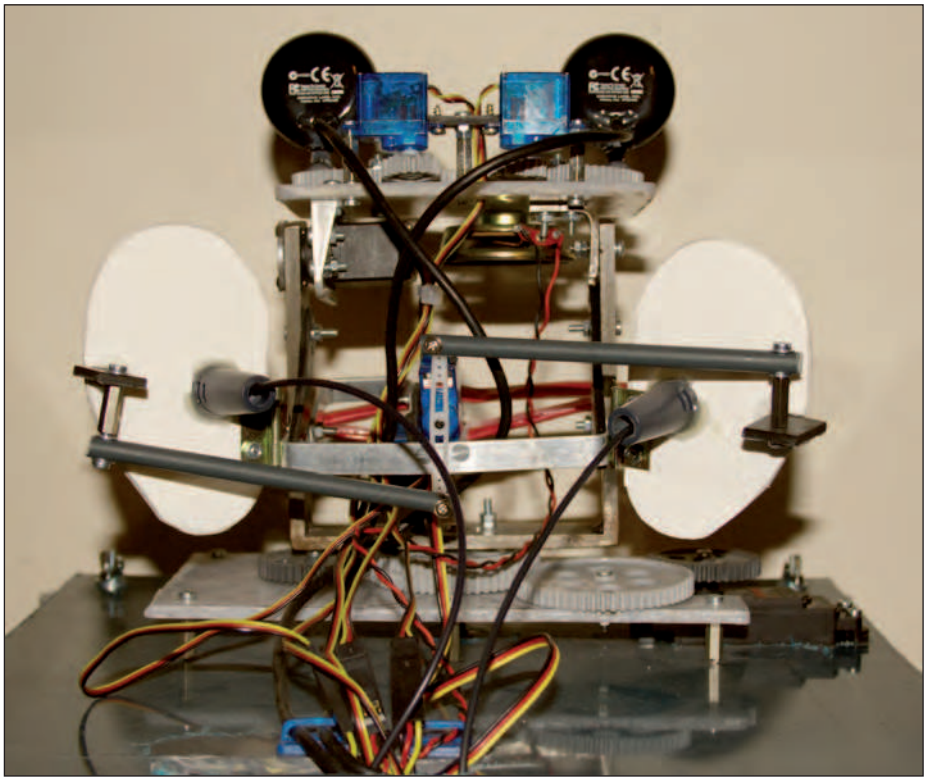


Figura 6: le orecchie sono mosse da un sistema biella-manovella direttamente da un unico servomotore.

creare un ciclo che apra e chiuda una porta del PIC per creare un appropriato treno di impulsi:

TEMPO DI APERTURA / TEMPO DI UN CICLO = CICLI NECESSARI

Questo PIC ha un TCY pari a 4/FOSC. Questo significa che idealmente con un clock di 20 MHz impiegherà 0,2 usec per compiere una singola istruzione, dunque avremo:

-90° = (600 / 0,2)usec = 3000 cicli
+90° = (2100 / 0,2)usec = 10500 cicli
refresh = (20ms / 0,2)usec = 100000 cicli

questi saranno i valori di partenza tramite i quali svilupperemo la nostra prima libreria software.

LA COSTRUZIONE DELLA TESTA

Il sistema di visione del Golem si basa su due webcam montate su un classico sistema di movimentazione "pan & tilt".
La parte centrale della faccia del Golem è composta da una forma a "C" in ferro che misura 110 mm per lato. Questa struttura ha il compito di supportare tutti gli altri elementi che poi verranno montati sopra. Per questo motivo deve essere abbastanza robusta ed esente da vibrazioni.
Questa parte rappresenta l'asse "pan" del sistema di visione artificiale ed è stata installata su una ruota dentata da 57 denti che poggia su un piano rialzato. Questo ingranaggio è congiunto ad un altro ingranaggio da 38 denti mosso da un potente servomotore Hitec HS755HB (chiamato *servomotore A*) che eroga una coppia di oltre 13 Kg/cm.
Questa potenza di azionamento unita ad un ulteriore rapporto di riduzione, riesce a muovere verso la posizione desiderata tutto il sistema testa del Golem, inoltre ne permette futuri upgrade.

IL BLOCCO DELLE WEBCAM

La testa è stata sviluppata principalmente per poter dotare il Golem di un'appropriata visione artificiale e, nel caso specifico, mi sono servito di una coppia di webcam Creative VF0330 aventi una risoluzione di 800x600 pixel. I sensori non

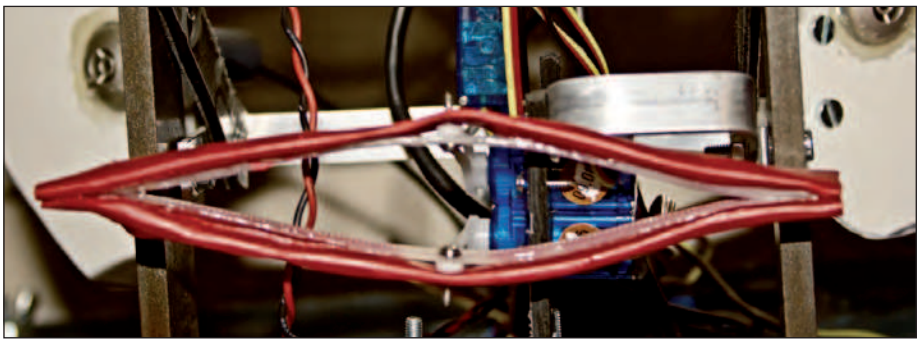


Figura 7: particolare delle labbra e della bocca del robot, i motori sono situati nella parte sinistra della testa.

sono stati bloccati in posizione fissa, bensì hanno la possibilità di muoversi sull'asse X grazie a delle ruote dentate di 58 denti sopra le quali sono stati montati. Questi ingranaggi sono accoppiati ad altri ingranaggi da 17 denti mossi da due piccoli servomotori Hitec HS55 (detti rispettivamente servomotore C e servomotore D).
Grazie all'ulteriore rapporto di riduzione 1:3 è possibile gestire indipendentemente la movimentazione delle webcam con una certa precisione che sarà necessaria durante la fase meccanica di convergenza/divergenza per la valutazione della distanza degli oggetti.
Momentaneamente il focus è manuale ma ben presto sarà supportato elettromeccanicamente per una maggiore automatizzazione del processo visivo mentre la distanza massima tra le due webcam è di circa 65 mm, il che rispecchia la distanza media interpupillare di un essere umano adulto.

Il piano delle webcam rappresenta il piano "tilt" del sistema visivo e viene gestito da un servomotore Hitec HS422 (chiamato servomotore B) che possiede una coppia di oltre 4 Kg/cm. Nel lato inferiore del piano, trova posto una piccola cassa acustica che doterà di voce il robot Golem.

LE ORECCHIE

Il robot è dotato di due microfoni per il rilevamento dei suoni, essenziali per l'interfacciamento con gli esseri umani. Per poter eliminare i molesti rumori di fondo che potrebbero ingannare il sistema di riconoscimento vocale e per poter meglio localizzare la fonte sonora, i sensori sono stati inseriti in un piano con delle "orecchie" realizzate in PVC e sono state fissate sulla struttura centrale grazie a delle cerniere che poggiano su un supporto di alluminio agganciato alla struttura centrale.
Questa composizione dona alle orecchie un grado di libertà pari ad un arco di 30° sul piano X che, oltre a rendere orientabili le stesse verso una fonte sonora, le rende anche simpatiche all'interlocutore. L'orientamento delle orecchie è stato reso possibile grazie ad un servomotore Hitech HS-55 (chiamato servomotore E) collocato tra le due orecchie e collegato ad sistema di biella-manovella che trasforma il movimento rotativo della squadra, in una spinta orizzontale capace di spostare, senza indugio, contemporaneamente entrambe le orecchie. I microfoni dovranno essere poi collegati alla motherboard X86 posta all'interno del piano centrale.

LA BOCCA

Per la bocca sono state elaborate diverse soluzioni meccaniche. La prima ad essere stata sviluppata era costituita da due cornici in PVC che dovevano ricordare l'apparato mandibolare dell'essere umano, ma la movimentazione era abbastanza imprecisa e richiedeva potenze in gioco importanti contro il piccolo lavoro che doveva svolgere.
La seconda soluzione è stata analizzata da un punto di vista diverso: riprodurre solamente il movimento delle labbra. Sono bastati due servomotori Hitec HS55 di bassa potenza (che chiameremo servomotore F e servomotore G) per muovere il labbro superiore ed inferiore, due stringhe di plastica morbida e della gomma sottile di colore rosso per dare l'effetto delle "labbra". Il risultato è abbastanza interessante e di facile realizzazione.

I COLLEGAMENTI CON IL PIANO DELLA TESTA

Arrivati a questo punto, dovremo ritrovarci con molti fili penzolanti che, ovviamente, devono essere collegati.
I due cavi USB delle webcam ed i due cavi audio con jack dei microfoni, dovranno essere collegati alla motherboard posta nel piano centrali del robot.
L'elenco che segue mostra tutte le connessioni che devono essere effettuate con la scheda Freedom I posta all'interno del modulo testa:

- il cavo del servomotore A deve essere collegato al connettore SV8

PER *approfondire...*

Sito ufficiale del robot Golem: http://www.robot-golem.org

PIC 18F4580: http://www.microchip.com/wwwproducts/Devices.aspx?dDocName=en020615

MPLAB IDE: www.microchip.com/mplab/

MPLABX IDE: www.microchip.com/mplabx/

C18 compiler:

http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=1406&dDocName=en010014&part=sw006011

Schedulatore dei processi: http://it.wikipedia.org/wiki/Scheduler

Sistemi operativi Real Time: http://it.wikipedia.org/wiki/Sistema_operativo_real-time

- il cavo del servomotore B deve essere collegato al connettore SV7
- i cavi dei servomotori C e D devono essere rispettivamente collegati ai connettori SV5 e SV4
- il cavo del servomotore E deve essere collegato al connettore SV6
- i cavi dei servomotori F e G devono essere collegati ai connettori SV3 e SV2

tutti i cavi dei servomotori, essendo abbastanza corti, sono stati allungati con una prolunga da 150 mm con terminali maschio/femmina.

INTRODUZIONE AL SOFTWARE

E' stata creata una piccola libreria per testare e configurare le varie parti elettromeccaniche che permettono alla testa del Golem di muoversi come noi vogliamo. Ovviamente, la libreria proposta non è quella definitiva, ma per ora si limita a gestire il sistema in modo poco pratico, che in gergo informatico viene denominato "busy waiting" (attesa attiva), questa comporta che mentre stiamo operando su una parte specifica della testa non possiamo agire in parallelo sulle altre, ma dovremo aspettare che la funzione termini la propria esecuzione. Ciò non toglie che il risultato finale possa avere un buon impatto visivo. Tutto il software del Golem è stato sviluppato per microcontrollori Microchip PIC utilizzando il tool MPLAB IDE col supporto del compilatore C18 Student Edition. Ultimamente sto utilizzando il nuovo Ide MPLABX che, basato sul più navigato Netbeans, è disponibile in versione beta per l'installazione sia per Windows che per GNU/Linux.
La libreria (servoBusyWaiting) è disponibile per il download sul sito www.farelettronica.com.
All'inizio della libreria sono stati definiti dei range (intervalli) di movimento che corrispondono alla posizione dei servomotori allo stato di riposo. Ad ogni funzione viene impostato il massimo arco di rotazione meccanica del servo. I range, inoltre, avranno la funzione di indicarci lo stato del servo, determinandone così la posizione assoluta rispetto agli assi di movimento (infatti, questi verranno incrementati o decrementati a seconda del movimento del servomotore).
Per utilizzare questa libreria relativo alla testa, è sufficiente inizializzare il classico file main.c avendo cura di settare accuratamente i registri del microcontrollore e ricordando di settare tutti i registri PortD come uscita nella fase di inizializzazione del programma. Se, ad esempio, volessimo muovere le orecchie del Golem, dovremo inserire il richiamo della funzione che gestisce le orecchie:

```
int i = 0;
// tramite un ciclo muovo le orecchie avanti ed indietro per 10 volte consecutive
for(i  = 0 ; i < 10 ; i++){
 // porta le orecchie in avanti
 moveEars('B', 2){
 // ritardo tra i 2 tipi di movimento
 Delay10KTCYx(100);
 // porta le orecchie indietro
 moveEars('F', 2){
}
```

Nelle prossime versioni il software sarà notevolmente migliorato, grazie ad un motore di scheduling interno per gestire al meglio i vari processi concorrenti. Il metodo più efficace per farlo, è creare un motore real time, utilizzando gli interrupt disponibili nei PIC come il Timer1.
Alcune operazioni vitali, come la comunicazione tramite il protocollo CAN con gli altri piani del robot, dovranno avvenire in un lasso di tempo accettabile dal sistema e, per fare questo, i servizi devono essere svolti entro un tempo limite massimo prestabilito, onde evitare malfunzionamenti a causa dello sfasamento delle risposte dei vari moduli. Sarebbe un peccato se il robot andasse a sbattere contro un muro a causa della scheda che gestisce le ruote poiché era occupata ad elaborare altro piuttosto che i segnali provenienti dal bus!

CONCLUSIONE

Nonostante la realizzazione di una testa particolarmente articolata non sia alla portata di tutti gli hobbisti, certo non mancano materiali e mezzi che ne permettono la costruzione senza necessariamente essere provvisti di macchinari di precisione o di attrezzature costose. Probabilmente i lettori che vorranno cimentarsi nella costruzione, difficilmente vedranno il loro lavoro concluso nell'arco di poco tempo ma non si devono scoraggiare: sicuramente il risultato finale li soddisferà.
Attualmente tutto il software è in fase "alpha" di sviluppo, il che lo rende del tutto inadatto ad un prematuro rilascio, dato che alcune funzionalità potrebbero cambiare nel breve periodo, anche se la struttura hardware è ormai definita. La fase beta del software dedicata al testing, ma già completa nelle sue funzionalità, sarà disponibile a breve e verrà pubblicata sul sito ufficiale del progetto: http://www.robot-golem.org
Nel prossimo articolo inizierò a descrivere il sistema di movimentazione del Golem, i materiali utilizzati e il perché sono stati utilizzati.
Valuteremo i sistemi di movimentazione elaborati inizialmente e le loro differenze, positive e negative. Posso già annunciarvi in anteprima che sono stati scelti componenti difficilmente reperibili nella robotica amatoriale, che hanno provocato non poche sorprese.

RINGRAZIAMENTI ED AVVISI

Noto con piacere che alcuni lettori hanno chiesto ulteriori informazioni sul progetto. Se anche voi avete domande, perplessità, critiche, se volete far parte del progetto o, ancora, siete semplicemente curiosi, visitate il sito ufficiale: http://www.robot-golem.org o contattatemi via mail a: info@robot-golem.org, sarò felice di rispondere a tutti i vostri quesiti. ❑

CODICE MIP 2804839

PROMOSSO DA:

assodel

Associazione Nazionale Fornitori Elettronica

I CONTENUTI

L'evento dedicato alla componentistica e ai sistemi per il fotovoltaico

Un incontro per tutta la supply chain sullo stato dell'arte del mercato italiano, dove approfondire aspetti di carattere normativo, tecnologico e applicativo

IL TARGET

Manager, ingegneri, progettisti, installatori, impiantisti, utenti, architetti, centri design, studi di progettazione, P.A.

IL FORMAT

Un **one-day event** articolato su interventi e convegni selezionati dal Comitato Tecnico Assodel
Una **community area** di incontro tra la domanda e l'offerta strutturata per fornire aggiornamenti tecnologici e novità di prodotto

A completamento **Educational**: seminari formativi di contenuto applicativo-progettuale

CODICE MIP 2799880

FORTRONIC
an assodel electronics forum

29 SETTEMBRE 2011

IV EDIZIONE

MILANO

CENTRO CONGRESSI MILANOFIORI
Strada 1 MilanoFiori - 20090 Assago (MI)

SOLAR
PV TECHNOLOGIES
FORTRONIC
an assodel electronics forum

event promoter

Solar & Lighting Revolution Farnell

partecipazione gratuita, previa registrazione

www.solarfortronic.it

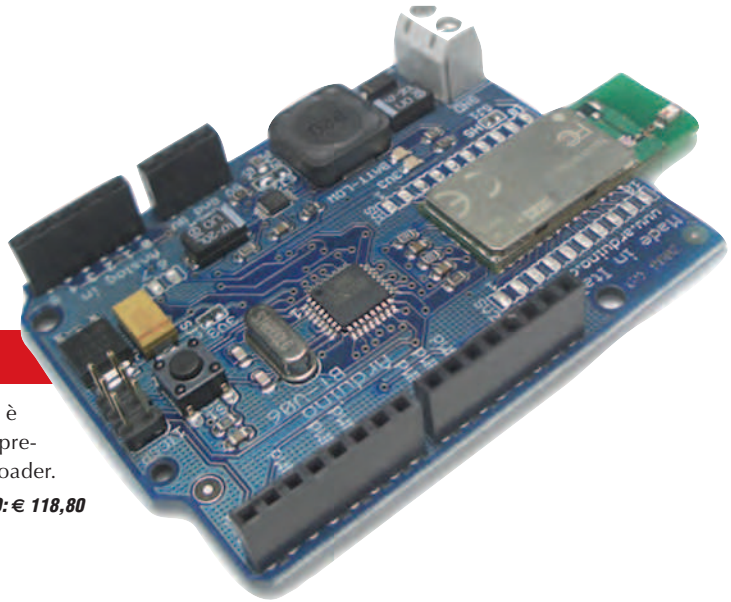
Informazioni e segreteria: Tel 02 210 111 236
e-mail: marketing@fortronicforum.com

ORGANIZZAZIONE:

TECNO IMPRESE

Via Console Flaminio 19 - 20134 Milano
Tel 02 210.111.1 • Fax 02 210.111.222
cons@tecnoimprese.it • www.tecnoimprese.it

Lo shop dei prodotti Inware Edizioni...



Connessione Bluetooth per Arduino

PRIMO PIANO

Arduino BT è la versione Bluetooth della scheda Arduino. Il modulo Bluetooth è un modulo classe 1 (WT11) per una distanza di trasmissione di 100m. La board è venduta completamente assemblata e testata con ATMega328 pre-caricato col bootloader.

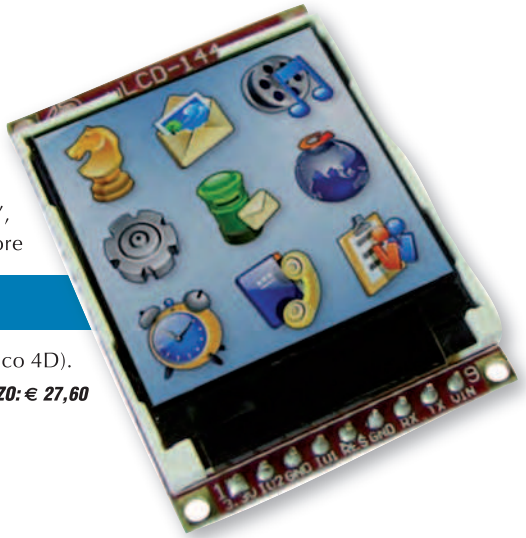
CODICE: A000002 PREZZO: € 118,80

NOVITA'

MOACON: modulo CPU

Modulo CPU MOACON con Processore ARM 32-bit (ARM CORTEX) a 72MHz. **Caratteristiche:** Processore ARM 32-bit (ARM CORTEX) - Frequenza di Clock: 72MHz - Memoria programma: 512KB Flash Memory - Memoria dati: 64KB SRAM -Memoria non-volatile: 32KB F-RAM -Built-in Real-Time Clock e Batteria -Porta RS232 Built-in -LCD e 7-Segment Control Port -Alimentazione 24V (Regolatore di potenza built-in)

CODICE: DP-CPU500 PREZZO: € 132.00

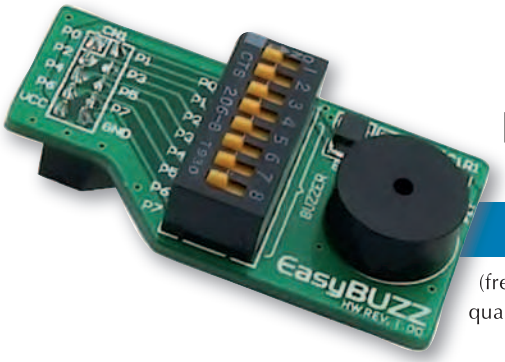


Display LCD a 65000 colori

uLCD-144 è un modulo display LCD TFT da 1,44", risoluzione 128x128 pixel, 65K colori, con processore grafico Goldelox embedded. Il modulo è disegnato per operare sotto due differenti piattaforme: comando seriale SGC o GFX (Linguaggio Grafico 4D).

NOVITA'

CODICE: ULCD-144 PREZZO: € 27,60



Buzzer per easyPIC

EasyBuzz è una board d'estensione che aggiunge un Buzzer piezoelettrico a qualsiasi Board di sviluppo Mikroe. Il buzzer è capace di riprodurre suoni nel range da 20Hz a 20KHz, ma la miglior prestazione è data dal range delle frequenze tra 2KHz e 4KHz (frequenza risonante 3.8KHz). EasyBuzz include un dip-switch per selezionare quale microcontrollore I/O dovrebbe essere usato per pilotare il buzzer.

NOVITA'

CODICE: EP-EASYBUZZ PREZZO: € 9.00

Telecamera seriale

NOVITA'

uCAM529 (microCAM) è un modulo videocamera seriale altamente integrato che può essere collegato a qualsiasi sistema host richiedente una videocamera o una videocamera a compressione JPEG per applicazioni imaging embedded. Il modulo utilizza un sensore colore VGA OmniVision CMOS insieme a un chip per la compressione JPEG che fornisce un sistema di videocamera a basso costo energetico ed economico. Il modulo è provvisto di un'interfaccia Seriale on-board (livelli TTL) ideali per una connessione diretta a qualsiasi micro-controllore host UART.

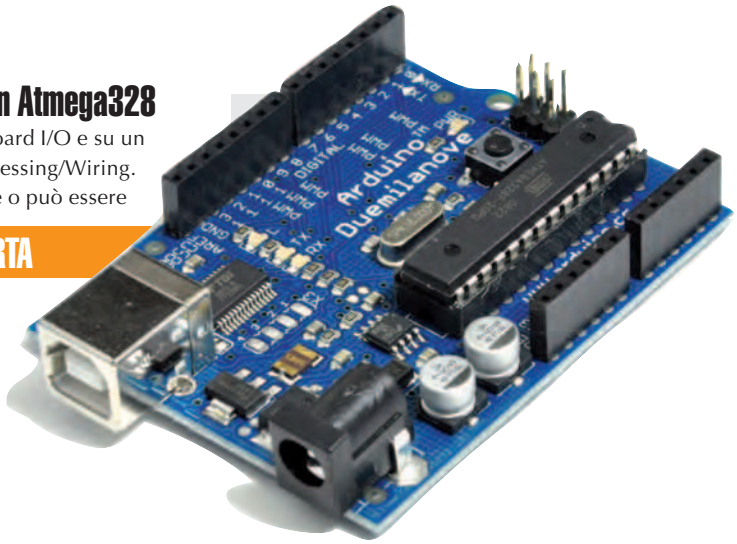
CODICE: UCAM529-TTL PREZZO: € 54.00

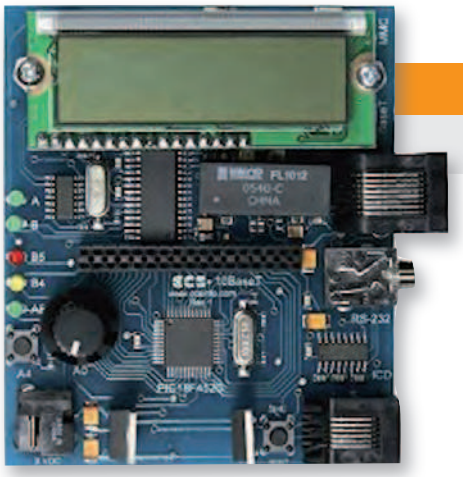
Arduino Duemilanove con Atmega328

OFFERTA

Arduino è una piattaforma fisica open-source basata su una semplice board I/O e su un ambiente di sviluppo user-friendly che implementa il linguaggio Processing/Wiring. Arduino può essere utilizzato per sviluppare oggetti interattivi stand-alone o può essere connesso al software presente sul vostro computer (ad esempio Flash, Processing, MaxMSP, PD). Duemilanove è la release migliorata della versione Diecimila. Aggiunge alle precedenti nuovi funzionalità (funzione autoreset, connettore di alimentazione esteso, LED built-in, protezione sovraccarico USB) la selezione di alimentazione automatica. La board è venduta completamente assemblata e testata col nuovo processore ATMega328 pre-caricato con bootloader.

CODICE: A000001 PREZZO: ~~€ 34.80~~ € 27.60





OFFERTA

Sistema di sviluppo Ethernet

Scheda di sviluppo basata sul chip Ethernet Microchip ENC28J60. Il chip è un controllore SPI, 10Mbit/sec e transceiver ethernet full duplex. La scheda contiene a bordo: un potenziometro, un pulsante, 3 LED, un convertitore di livello RS232 e un PIC18F4520. Inoltre sono presenti un display LCD, una EEPROM seriale e un lettore di card MMC/SD.

CODICE: MPK-ETH PREZZO: ~~€ 130.80~~ € 02.00

BEST SELLER

6 pubblicazioni digitali per te

Un bundle di 6 FE Digital Extra a un prezzo incredibilmente conveniente. Il bundle comprende i seguenti fascicoli in formato pdf: "100 progetti per 1000 applicazioni", "200 progetti per 2000 applicazioni", "AVR projects", "Costruisci la tua CNC", "PIC Projects vol.1", "PIC Projects vol.2"

CODICE: BUNDLE DE PREZZO: € 9.90

Lo shop dei prodotti Inware Edizioni... ma non solo!

Usa IEMOBILE!

Sottoscrivi un abbonamento CLUB al prezzo del PREMIUM! L'abbonamento CLUB prevede 6 numeri di Firmware Edizione Digitale, compreso 1 anno di iscrizione al Club di Firmware e 1 anno di Fare Elettronica

Usa IEmobile! Abbonamento CLUB Firmware digitale al prezzo del premium!"

BUNDLE

Un tool per programmare i PIC in C!

Il compilatore MikroC e l'eBook per imparare ad usarlo a un prezzo speciale! Con questi strumenti potrai imparare a programmare i PIC in linguaggio C procedendo con esempi pratici.

CODICE: BUNDLE U PREZZO: € 79.90

Sviluppare con ATmega128

La mikroXMEGA Board è una board prototipo di piccole dimensioni per lo sviluppo efficiente di dispositivi che utilizzano il nuovo microcontrollore Atmel® XMEGA™ A1.

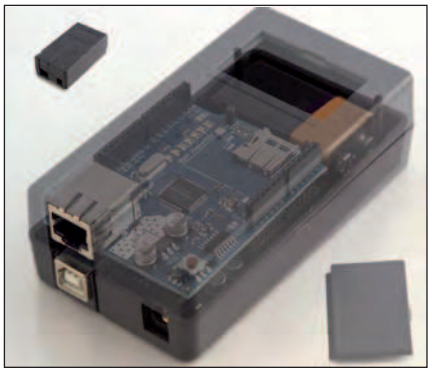
La board contiene il dispositivo ATxmega128A1 che è connesso a un circuito oscillatore a 8MHz.

CODICE: MIKROXMEGA

PREZZO: € 23.88

Box per Arduino

Box per Arduino: Box di Plastica (ignifuga) che può alloggiare una scheda Arduino



(Duemilanove - Diecimila - NG o MEGA), uno shield ETH e una batteria. Prodotto in ABS / plastica. Dimensioni 87x64x28mm. Tutte le parti interne (board e batteria) non sono incluse.

CODICE: A000009

PREZZO: € 11.88

Il LED e l'illuminazione

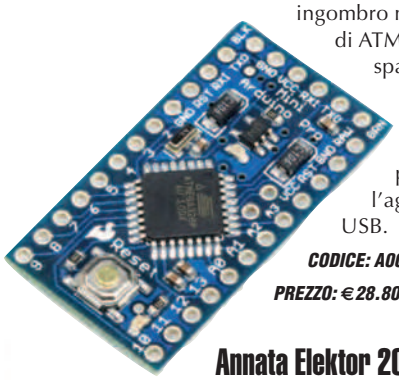
Partendo dalle nozioni basilari sulla luce e proseguendo con la teoria dei semiconduttori e l'optoelettronica, questo volume spiega cosa sono e come si accendono i led, proponendo schemi applicativi per utilizzarli sia come semplici spie che nelle realizzazione di lampade per illuminazione di ambienti. Non mancano utili note di illuminotecnica e calcoli esemplificativi.

CODICE: FE-34

PREZZO: € 11.00

Arduino Mini

Versione miniaturizzata delle scheda Arduino (senza parte USB). Le dimensioni estremamente ridotte (30x18mm) permettono una riduzione dello spazio di ingombro non indifferente. Fa uso di ATMega168 con 16K di spazio programma ed è utilizzabile con convertitore USB/Seriale per la programmazione e l'aggiunta di una porta USB.



CODICE: A000003

PREZZO: € 28.80

Annata Elektor 2010

La raccolta delle annate di Elektor in italiano uscite nel 2010. In totale oltre 10 riviste in PDF ad alta risoluzione, potrai stampare gli articoli di tuo interesse senza perdere la qualità della rivista.

CODICE: EK-CD2010

PREZZO: € 9.90

prossimamente su Fare Elettronica

COMING...

COME ALLESTIRE UNA STAZIONE RADIOAMATORIALE

Una serie di utili consigli per chi intende allestire una stazione radio in casa propria ed entrare a far parte del mondo dei radioamatori.

CRONOMETRO SEGNATEMPO

La realizzazione di un cronometro, utilizzando il microcontrollore Pic. Pratica e didattica per lavorare al meglio con il compilatore mikroBASIC PRO.

ESPERIMENTI CON I DIODI

La giunzione p-n è la struttura fisica fondamentale della maggior parte dei dispositivi elettronici allo stato solido. Il diodo a giunzione è il componente più semplice realizzabile con una struttura di questo tipo. In questo articolo ne illustriamo, a scopo didattico, le applicazioni circuitali più comuni attraverso una serie di semplici esperimenti pratici.

Ordinare direttamente tramite internet conviene! Potrai infatti aggiudicarti dei buoni sconto (coupon) con le modalità:

COUPON ABBONAMENTO ONLINE FE E FW

Sottoscrivendo un nuovo abbonamento o rinnovando il proprio online, direttamente tramite IE SHOP, si ha diritto ad un coupon del 15%! **Qualora ci si abbona (sempre online) a Fare Elettronica e Firmware insieme il coupon è del 20%!**

COUPON QUIZ "ELETTROQUIZ"

Rispondendo ai Quiz presente ogni mese su Fare Elettronica si avrà diritto ad un coupon del 10%. Tutte le risposte pervenute saranno accuratamente valutate dagli autori dei quesiti e, in caso di risposta esatta (oltre al Coupon) potrai vincere fantastici premi!

COUPON "ACQUISTI PREMIATI"

I tuoi acquisti su www.ieshop.it vengono premiati con un coupon del 10% se il tuo ordine supera i 100 EUR (IVA e spese di trasporto escluse). Ma se il tuo ordine supera i 200 EUR, lo sconto sale al 20%!

IMPORTANTE!

- Il coupon può essere speso **esclusivamente** ordinando sul sito www.ieshop.it
- Il coupon non potrà essere utilizzato in caso di nuova sottoscrizione o rinnovo dell'abbonamento, è invece utilizzabile per tutti gli altri prodotti presenti su www.ieshop.it
- Ogni coupon ha una scadenza, non dimenticarlo!
- Lo slogan "più acquisti più risparmi" pecca senz'altro di originalità, ma in questo caso è molto azzeccato: i coupon sono infatti utilizzabili solo una volta, usalo bene!
- I coupon non sono cumulabili, e vanno utilizzati singolarmente, secondo il criterio cronologico di assegnazione.

CONTI CORRENTI POSTALI - Ricevuta di Versamento — BancoPosta

sul C/C n. 70107552

di Euro

importo in lettere

INTESTATO A:
INWARE EDIZIONI SRL

ESEGUITO DA:

VIA - PIAZZA

CAP

LOCALITÀ

BOLLO DELL'UFF. POSTALE

AVVERTENZE
Il Bollettino deve essere compilato in ogni sua parte (con inchiostro nero o blu) e non deve recare abrasioni, correzioni o cancellature.
La causale è obbligatoria per i versamenti a favore delle Pubbliche Amministrazioni.
La mancata, parziale o inesatta indicazione delle informazioni richieste in modo leggibile, in relazione alle finalità per le quali si compone il bollettino.

CAUSALE

CONTI CORRENTI POSTALI - Ricevuta di Accredito — BancoPosta

sul C/C n. 70107552

di Euro

importo in lettere

TD 451

INTESTATO A:
INWARE EDIZIONI SRL

CAUSALE

ESEGUITO DA:

VIA - PIAZZA

CAP

LOCALITÀ

BOLLO DELL'UFF. POSTALE
codice bancoposta

IMPORTANTE: NON SCRIVERE NELLA ZONA SOTTOSTANTE
importo in euro — numero conto — td

AUT. DB/SISB/E 20017 DEL 11.04.2006

70107552 < 451>

IMPORTANTE: NON SCRIVERE NELLA ZONA SOPRASTANTE!

AVVERTENZE

Il Bollettino deve essere compilato in ogni sua parte (con inchiostro nero o blu) e non deve recare abrasioni, correzioni o cancellature.

La causale è obbligatoria per i versamenti a favore delle Pubbliche Amministrazioni.

Le informazioni richieste vanno riportate in modo identico in ciascuna delle parti di cui si compone il bollettino.

Indagine sui lettori

Aiutaci a conoscerti meglio!
Con il tuo aiuto riusciremo ad offrirti una rivista sempre più in linea con le tue aspettative!

Nome .....

Cognome .....

Via ..... n° .....

Cap ..... Città ..... Prov .....

Email .....

IL TUO SETTORE DI COMPETENZA:

- ❑ **B05** Direzione Tecnica
- ❑ **B06** Progettazione
- ❑ **B07** Studente
- ❑ **B08** Direzione Acquisti
- ❑ **B09** Insegnante
- ❑ **B10** Altro . . . . . .

PRODOTTO PRINCIPALE O SERVIZIO OFFERTO DALL'AZIENDA DOVE LAVORI:

- ❑ **C11** Apparecchiature elettriche, elettroniche, ICT
- ❑ **C12** Elettrodomestici
- ❑ **C13** Consulenza
- ❑ **C14** Apparecchiature scientifiche, misura e controllo
- ❑ **C15** Automotive
- ❑ **C16** Vending
- ❑ **C17** Altro . . . . . .

NUMERO DI DIPENDENTI DELLA TUA AZIENDA:

- ❑ **D18** fino a 10
- ❑ **D19** da 10 a 50
- ❑ **D20** da 50 a 100
- ❑ **D21** da 100 a 500
- ❑ **D22** oltre 500

FE 312

Solo se sei abbonato, indica il tuo codice abbonato: . . . . . .
e barra la casella di interesse:

TIPO DI ABBONAMENTO:

- ❑ **A01** Personale uso professionale
- ❑ **A02** Aziendale
- ❑ **A03** Scuola o Università
- ❑ **A04** Personale uso hobbistico

Compila il cedolino e invialo in busta chiusa o via fax allo 02 66508225 e riceverai GRATIS a tua scelta UN CD-ROM DEL VALORE DI 10 EURO

STARTER KIT
per lo sviluppo di applicazioni con i microcontrollori Freescale 9RS08

MASTERIZZARE DVD E FORMATO DIVX
un corso che spiega con esempi pratici come masterizzare DVD e in formato DivX.

Grazie per la preziosa collaborazione!

PRODOTTI

Se vuoi ordinare "offline" ovvero senza usare internet, usa il seguente modulo d'ordine **PRODOTTI**

COD. PRODOTTO	DESCRIZIONE	PREZZO UNITARIO	Q.tà	Totale
			SUB TOTALE	
			SPESE DI SPEDIZIONE	
			TOTALE	

Nota. Se le righe non sono sufficienti, utilizzare più copie del modulo.
Per maggiori dettagli sulle spese di spedizione e i metodi di pagamento vedi retro cartolina.

METODI DI PAGAMENTO

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **CONTRASSEGNO**

FE 312

☐ MasterCard ☐ VISA ☐ postepay

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, Via Giotto,7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . .
Cap . . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . .
Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

ABBONAMENTO

Sì, mi abbono a Fare Elettronica

FE 312

☐ Abbonamento TRIAL a 3 numeri di FARE ELETTRONICA a soli € 16,50 anzichè € 18,00
☐ Abbonamento PREMIUM a 11 numeri di FARE ELETTRONICA a soli € 55,00 anzichè € 66,00
☐ Abbonamento PRO a 11 numeri di FARE ELETTRONICA include il Cd dell'annata 2010 a soli € 59,50 anzichè € 96,00

METODI DI PAGAMENTO (per maggiori dettagli vedi retro cartolina)

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **ALLEGO ASSEGNO** (Intestato a Inware Edizioni)

☐ MasterCard ☐ VISA ☐ postepay

PayPal fare il pagamento su paypal@inwaredizioni.it

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, Via Giotto,7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . .
Cap . . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . .
Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

ISTRUZIONI PER L'ORDINE E SPESE DI SPEDIZIONE

Il modulo d'ordine dovrà essere compilato in tutte le sue parti ed inviato via posta o via fax ai recapiti indicati sul modulo d'ordine stesso. Gli ordini potranno essere fatti anche direttamente online dal sito www.ieshop.it. In questo caso non sarà necessario inviare il modulo d'ordine. Le spese di spedizione ammontano a euro 8,50 a cui vanno aggiunti euro 3,50 se si sceglie di pagare in contrassegno.

METODI DI PAGAMENTO

Si accettano pagamenti ino in contrassegno, carta di credit, bollettino postale o bonifico all'ordine. Per il contrassegno verrà applicata una spesa aggiuntiva di euro 3,50 per le spese di contrassegno. Forme diverse di pagamento devono essere previamente concordate.

Bonifico bancario

Appoggiato su Poste Italiane IBAN: IT 68 I 07601 01600 000070107552 intestato a Inware Edizioni srl.

Bollettino postale

Versamento sul conto corrente postale n. 70107552 intestato a Inware Edizioni srl. Inviare la ricevuta (o copia) del versamento.

Contrassegno

La merce verrà pagata direttamente al corriere alla consegna della merce. Il pagamento in contrassegno comporta l'addebito di euro 3,50per spese di contrassegno.

Carta di Credito

Il pagamento con carta di credito può essere effettuato specificando i dati della carta via posta, email, web, fax o telefono. Per gli ordini effettuati sul sito il pagamento mediante carta di credito verrà effettuato attraverso una connessione sicura SSL per garantire la massima sicurezza.

TERMINI E CONDIZIONI

Tutti i prodotti sono coperti da garanzia. La garanzia copre tutti i difetti di fabbricazione entro un anno dal ricevimento della merce. Tutti i prodotti non funzionanti per uso improprio o incuria non saranno ritenuti in garanzia e saranno addebitati gli eventuali costi di riparazione. Tutti i prodotti verranno riparati e/o sostituiti direttamente dal produttore. Non sono coperti da garanzia i componenti elettronici (microprocessori, memorie, ecc.) La garanzia dei prodotti si intende F.co ns. sede, le eventuali spese di trasporto sono a carico del cliente salvo accordi diversi. Per dar corso alla riparazione/sostituzione in garanzia è necessario seguire l'apposita procedura di RMA.

PRIVACY

Ai sensi del Decr. Lgs. 196/2003 la informiamo che i dati trasmessi verranno impiegati coi principali scopi di indagini di mercato e nelle modalità previste dallo stesso, prevalentemente con mezzi informatici. Il conferimento, di norma facoltativo, è obbligatorio per permettere il rapporto commerciale. È in ogni caso fatto diritto dell'interessato esercitare i propri diritti, nei modi previsti dal "Titolo II art. 7" della legge sopra citata, scrivendo a Inware Edizioni srl, Via Cadorna 27 – 20032 Cormano o tramite email all'indirizzo info@inwaredizioni.it

---

ABBONARSI ALLE RIVISTE INWARE EDIZIONI CONVIENE!

I vantaggi per gli abbonati sono, oltre al prezzo bloccato per un anno, la ricezione del numero direttamente a casa con la garanzia di ricevere tutti i numeri. Inoltre un vistoso risparmio che, nel caso dell'abbonamento PLUS, ammonta a ben 36,50 euro. L'abbonamento ha una durata di 12 mesi e comporta l'invio di 11 numeri di Fare Elettronica. Eventuali variazioni di indirizzo andranno comunicate tempestivamente alla Redazione che provvederà a registrare il cambiamento senza alcuna spesa aggiuntiva. L'abbonamento decorrerà dal primo numero raggiungibile alla data di avvenuto pagamento. Non sono previsti rimborsi in caso di disdetta dell'abbonamento.

METODI DI PAGAMENTO

Bonifico bancario

Appoggiato su Poste Italiane IBAN: IT 68 I 07601 01600 000070107552 intestato a Inware Edizioni srl.

Bollettino postale

Versamento sul conto corrente postale n. 70107552 intestato a Inware Edizioni srl. Inviare la ricevuta (o copia) del versamento indicando nella causale: "Abbonamento Fare Elettronica"

Carta di Credito

Il pagamento con carta di credito può essere effettuato specificando i dati della carta via posta, email, web, fax o telefono. Per gli ordini effettuati sul sito il pagamento mediante carta di credito verrà effettuato attraverso una connessione sicura SSL per garantire la massima sicurezza.

Assegno bancario

E' possibile spedire un assegno bancario insieme a questo coupon.

Paypal

Inviare il pagamento a paypal@invaredizioni.it.

Hai un quesito di elettronica?
Vuoi comunicare con altri appassionati?
Vuoi condividere le tue esperienze?
Tutto questo su IEforum!
www.ieforum.it
CODICE MIP 2808409

Il primo oscilloscopio mixed-signal low-cost!

Oscilloscopio, Analizzatore di spettro

Analizzatore Logico, Generatore Logico

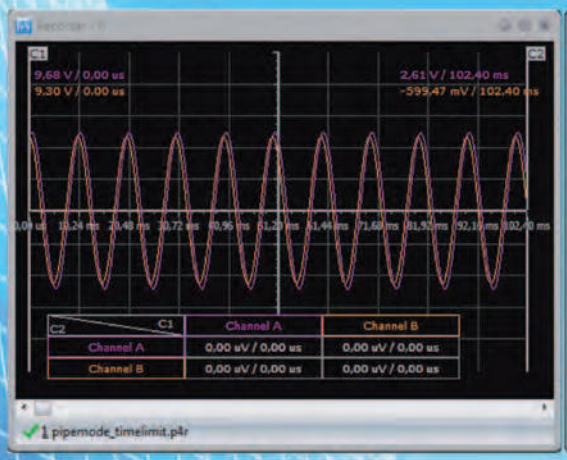
2+5 Canali analogici (12bit, 1MS/s)

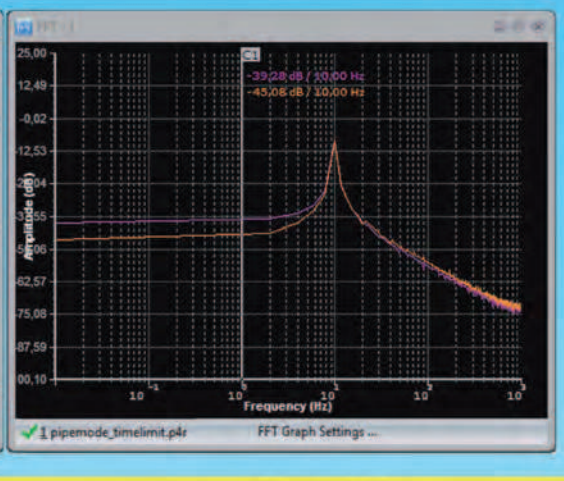
Generatore di onde quadre e PWM

Registratore Digitale

16 Canali Digitali

Aggiornamenti software gratuiti





Oscilloscopio ed analizzatore di spettro

Numero canali: 2+5
Frequenza di campionamento: fino a 1MHz
Memoria:
- Buffer di lettura: 1126 campioni/canale (1 canale), 563 campioni/canale (2 canali).
- Pipe di lettura: 64K campioni/canale (1 o 2 canali).

Massima tensione di ingresso: -20 ÷ +20 V
Risoluzione ADC: 12bits
Triggering:
- Assoluto (per fronti di salita/discesa)
- Differenziale (per differenza tra campioni consecutivi)
- Esterno (per fronti di salita/discesa di segnali TTL)

Funzionalità disponibili: Hamming, Hanning, Blackman, Blackman-Harris.

Analizzatore logico

Numero canali: 16 (8 se utilizzato il generatore logico)
Frequenza di campionamento: fino a 10MHz
Memoria:
- Buffer in lettura (Fs=4-8 MHz) 128 bit/canale.
- Buffer in lettura (Fs=2-2.66 MHz) 1160 bit/canale.
- Buffer in lettura (Fs<=1 MHz) 1544 bit/canale
- Buffer in lettura (in mod. concatenamento) 1 Mbit/canale.
- Pipe di lettura (Fs < 500KHz) 4K a 256 Mbit/canale.

Massima tensione di ingresso: 0 ÷ +5 V
Triggering: per fronti del segnale, maschere, impulsi persi, clock esterno.
Clock: interno/esterno

Registratore

Frequenza di campionamento: fino a 1MHz
Capacità massima di registrazione: 24 ore (Fs < 100 Hz)
Tensione d'ingresso: -20 ÷ +20 V (hardware 2 sub-band)
Risoluzione ADC: 12bits

Generatore logico

Numero canali: 8
Frequenza di campionamento: fino a 1MHz
Memoria: 4000 bit/canale
Tensione di uscita: "0" - 0 V, "1" - 3.3 V
Massima corrente in ingress/uscita: 10 mA

CODICE MIP 28084401

Novità! Il più economico isolatore USB!

PoScope mega1

Ordinali subito su www.ieshop.it/poscope